Report LR-758

B2000 User Manual for Crack Analysis, Solution Data Mapping and Error Calculation

April 1994

H.A.J. Knops

TU Delft
Delft University of Technology

Faculty of Aerospace Engineering

B2000 User Manual for Crack Analysis, Solution Data Mapping and Error Calculation

H.A.J. Knops

Contents

page

1. Introduction 3

2. Crack analysis 5
   2.1 Theoretical background 5
      2.1.1 Mode enrichment 5
         2.1.1.1 Basic principle 5
         2.1.1.2 The enriched element domain 7
         2.1.1.3 Corotational formulation 8
         2.1.1.4 The resulting nonlinear equations 13
         2.1.1.5 Solution of the corrector equations 15
         2.1.1.6 A posteriori mode enrichment 18
         2.1.1.7 Numerical integration 19
      2.1.2 The elastic T-term via mode enrichment 20
         2.1.2.1 Introduction 20
         2.1.2.2 Solution strategy 21
         2.1.2.3 Numerical procedure 22
   2.2 Interactive input commands 25
   2.3 Additional in- and output files 29
      2.3.1 Input file 'mode_enrich' 29
      2.3.2 Input file config 35

3. Error calculation 37
   3.1 Theoretical background 37
      3.1.1 The energy norm 37
      3.1.2 Modified version of the energy norm 41
   3.2 Interactive input commands 42
   3.3 Additional in- and output files 44

3.3.1 Output file 'ERRORS' 44

4. Solution data mapping 45

4.1 Theoretical background 45

4.1.1 Node projection 45

4.1.2 The mapping process 47

4.2 Interactive input commands 48

4.3 Additional in- and output files 49

4.3.1 Input file 'mode_enrich' 49

4.3.2 Input file 'config' 50

4.3.3 Input file 'configo' 50

5. Modifications for the load processor 51

5.1 Theoretical background 51

5.2 Interactive input commands 51

5.3 Additional in- and output files 51

5.3.1 Input file 'mode_enrich' 51

5.3.2 Input file 'config' 52

6. Examples 53

Literature 55

1 Introduction

The modular finite element package B2000 [17,18] is extended for the analysis of cracks in a geometrically nonlinear environment. For this purpose a number of routines that are already available in B2000 had to be adapted and some new numerical tools had to be developed. These crack analysis tools are implemented for the 4-, 8- and 9-node Bathe elements [1].
This report not only serves as a user manual for the special purpose routines that were developed but also contains information about the theoretical background of these routines. Chapters 2 to 5 deal with the separate routines that were adapted or developed and these chapters are written according to the following set-up:

section 1: Theoretical background
section 2: Interactive input commands
section 3: Additional in- and output files

In chapter 2 the routine for calculating the stress intensity factors and the elastic T-term will be dealt with. This routine actually represents an extended version of the continuation processor 'b2cont' and calculates the crack tip parameters (K_I, K_{II}, K_{III}, K_{BI}, K_{BII} and T) with the so-called mode enrichment technique. This mode enrichment technique is implemented in the nonlinear Bathe element and thus enables the calculation of the crack tip parameters for linear as well as geometrically nonlinear crack problems.
In chapter 3 the routine for calculating a global and local error over the total finite element mesh is discussed. Till now this routine can only be applied for the Bathe elements and it produces information that has to be available in order to perform adaptive mesh refinement. The actual refinement of a finite element mesh is not implemented in the B2000 finite element package but is written as a stand-alone program that can also be used in connection with other finite element packages. An input description for this mesh generator can be found in [16].
Chapter 4 deals with a routine for mapping the solutions data from one finite element mesh onto another. This solution data mapping technique is very useful for geometrically nonlinear problems where the nonlinear calculation after mesh refinement can be continued without having to reload the model again from the beginning. After regaining equilibrium again for the new grid layout the nonlinear calculation can be proceeded at the present load level. This

technique proves to be very important for crack growth simulations where one only has to open the new crack increments obtained from each crack growth step via a load relaxation method.

In chapter 5 finally some special adaptations for the load processor will be discussed briefly. These changes concern the fact that we want to deal with cylindrical shells and therefore have to be able to work in a cylindrical coordinate system. For this purpose the external loads, which are by default defined in a cartesian coordinate system have to be transformed to a cylindrical coordinate system.

This report does not contain any sample calculations concerning the results of the application of the developed B2000 routines. The interested reader is referred to [14,15] in order to see what the capabilities of this program are.

2 Crack analysis

2.1 Theoretical background

2.1.1 Mode enrichment

2.1.1.1 Basic principle

For a shell or a plate the general form of the linear elasticity solution for the stresses in the immediate surrounding of a sharply edged crack does not depend on the loads and boundary conditions applied to the structure at a finite distance of the crack tip. Only the stress intensity factors, which are characteristic parameters in this solution, depend on these conditions [5,20]. It is this property that is exploited by the singular mode superposition technique.
The general solution for a crack in a plate or a shell in terms of the displacements $\underline{u} = (u,v)^T$ and stresses $\sigma = (\sigma_{xx}, \sigma_{yy}, \sigma_{xy})$ tangential to the midsurface can be decomposed in two parts namely

$$\underline{u} = \underline{u}_1(r,\theta,z) + \underline{u}_2(r,\theta,z) \qquad (2.1a)$$

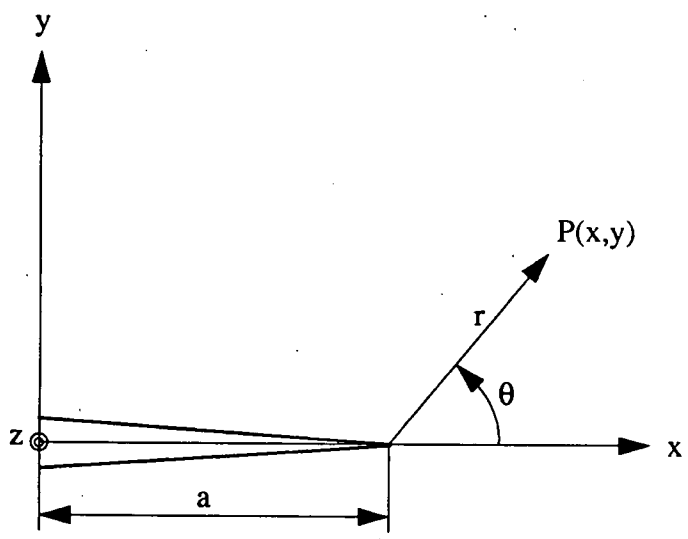


Figure 1: Straight crack in an infinite plate

$$\sigma = \sigma_1(r, \theta, z) + \sigma_2(r, \theta, z) \tag{2.1b}$$

where r and θ are the components of a polar coordinate system at the tip of the crack (as is depicted in Figure 1) and z is the distance measured along the normal **n** to the midsurface of the plate. The subscript (1) refers to the regular part of the solution and the subscript (2) refers to the singular crack tip solution that is dominant in a close neighborhood of the crack tip. When using a plate theory that includes transverse shear deformation there is also a tearing mode that is governed by an out of plane displacement field w and by transverse shear stresses $\sigma = (\sigma_{xz}, \sigma_{yz})$.

For a shell in a general state of deformation the in-plane crack tip solutions can be cast in the form

$$\underline{u}_2(r,\theta) = \sum_{i=1}^{2} \sqrt{r}\{K_i\Phi_i(\theta) + zk_iX_i(\theta)\} \tag{2.2a}$$

$$\sigma_2(r,\theta) = \sum_{i=1}^{2} \frac{1}{\sqrt{r}}\{K_i\Psi_i(\theta) + zk_i\Theta_i(\theta)\} \tag{2.2b}$$

where

$$K_i\,(i=1,2) = (K_I, K_{II})$$
$$k_i\,(i=1,2) = (k_1, k_2)$$

and the particular functions Φ_i, X_i, Ψ_i and Θ_i depend only on the specific shell theory that is used. If the transverse shear deformation is taken into account one also has

$$\underline{u}_{2III}(r,\theta) = w = \sqrt{r}\{K_{III}\Xi(\theta)\} \tag{2.3a}$$

$$\sigma_{2III}(r,\theta) = \frac{1}{\sqrt{r}}K_{III}Y(\theta) \tag{2.3b}$$

The stress intensity factors in these expressions are determined by the load intensity, the boundary conditions, the geometry and the physical properties of the structure surrounding the crack. The solutions (2.2a-b) and (2.3a-b) are generally applicable as long as the problem remains within the realm of the theory of linear elasticity.

It is now noted that the function $\underline{u}_2$ varies with $\sqrt{r}$ while the lowest order terms of $\underline{u}_1$ are proportional to r. For the stresses a similar qualification holds. The crack tip solutions σ_2 are proportional to $1/\sqrt{r}$ while the leading term in σ_1 is a constant. Due to these singular $1/\sqrt{r}$

terms the crack tip solution is dominated by $\underset{\sim}{u}_2$ and σ_2 while away from the crack the solution field is represented by the composition (2.1a-b). At the crack tip r = 0 all derivatives of $\underset{\sim}{u}_2$ with respect to r are singular, while away from the crack tip the solution is smooth with derivatives that are continuous in the entire remaining domain of the structure.

As noted [2, 8, 11, 12, 19, 25, 26, 28] the separation of the general solution into two parts, i.e. into a local singular field plus a regular global field can be carried over into a finite element solution scheme in an almost natural way. The first step in this approach is to construct a representation of the global field $\underset{\sim}{u}_1$ in terms of a regular finite element distribution. The second step is to add to this field (locally) the functions $\underset{\sim}{u}_2$ so that the completion of the solution is obtained. The result of such a discretization is that the solution is represented in terms of a nodal displacement set **q** belonging to the finite element distribution and a set **k** of stress intensity factors that are coupled with the local singular fields (2.2a-b) and (2.3a-b). In this way the stress intensity factors appear as an additional set of computational freedoms. In other words, this technique produces the stress intensity factors as an integrated part of the set of computational freedoms.

The displacement functions corresponding the membrane stress intensity factors do not depend on the kind of plate theory that one is using. However, for the bending (and tearing) stress intensity factors there is a difference between the displacement solutions according to the Kirchhoff [27] and the Reissner plate theory [9,10]. Because we are using a plate element that is based on a shear deformable plate theory we have to apply the analytical displacement solutions based on the Reissner plate theory. For further details the interested reader is referred to [14,15].

The extension of the range of applications of the method to geometrically nonlinear problems requires some careful consideration with regard to the formulation of the superposition process. In the following sections we will discuss three major aspects of this approach that need clarification. The first concerns the requirements that we have to impose on the size of the enriched element domain and on the continuity of the modes $\underset{\sim}{u}_1$ and $\underset{\sim}{u}_2$ combined. Another issue which needs attention is the implementation of the local crack tip solution in a geometrically nonlinear environment with large rotations and translations. Finally, because the governing equations are nonlinear and because it is advantageous to keep the required changes in the solution algorithm at a minimum, it is necessary to discuss the way the governing equations are solved in the altered situation.

2.1.1.2 The enriched element domain

The first aspect which needs attention is the size of the domain in which mode enrichment is applied. As discussed in [14,15] the crack tip solutions for the two bending modes and the tearing mode are only valid within a small boundary layer. Therefore, in contrast to the

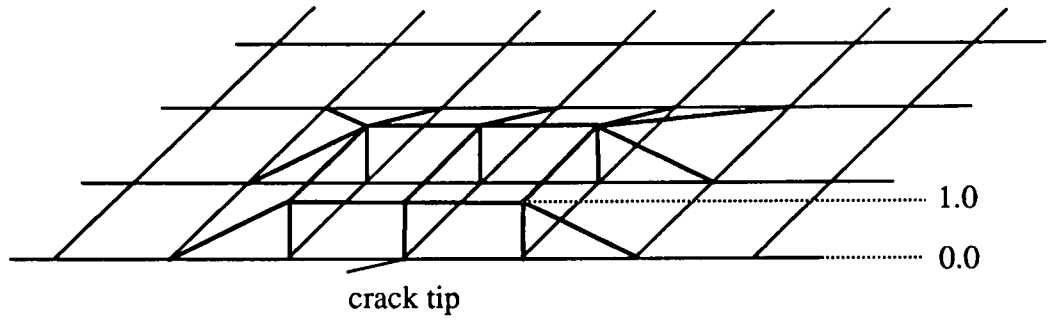


Figure 2: Definition of the bilinear transition function

membrane stress intensity factors, mode enrichment for the bending en tearing modes has to be restricted to a small boundary layer in order to obtain accurate results for these stress intensity factors. This is the main reason why the enriched element domain for the membrane modes on one hand and the bending and tearing modes on the other hand can be chosen independently.

Furthermore, we will restrict the mode enrichment process to some (small) area around the crack tip. In a strict sense one should require that the summation of $\underset{\sim}{u}_1$ and $\underset{\sim}{u}_2$ remains C_0-continuous everywhere in the domain. In practice this means that one should enforce this condition at the boundary of the enriched area. A simple way to accomplish this is to use a transition function. This function is equal to 1.0 for the enriched elements next to the crack tip. For the transition elements though this function is equal to 1.0 next to the enriched elements and equal to 0.0 next to the conventional elements (Figure 2) while the descent of this transition function is assumed to be bilinear. How to enter all this information in the B2000 routine 'b2contcrack' will be discussed in section 2.3.1.

The particular application of the mode enrichment technique that we have in mind is the problem of a longitudinal crack in a pressurized fuselage. This implies that we are enriching a curved surface with analytical crack tip solutions that only hold for a flat crack tip zone. However, this difficulty is circumvented by restricting mode enrichment to a small area around the crack tip which is approximately flat. Therefore the possible curvature of a structure plays no role as far as the mode enrichment process is concerned.

2.1.1.3 Corotational formulation

The set of singular functions discussed in section 2.1.1.1 is taken from the linear elasticity solution for an infinite plate with a crack under general loading conditions. It is defined with respect to the orientation of the undeformed crack geometry, which, in the linear theory, does not appreciably change during the loading phase. When the problem is geometrically nonlinear, as it is in our case, the crack front may undergo a finite displacement and rotation

under the given load. In that case the set of shape functions $\underline{u}_2$ should be defined with respect to the rotated geometry. Consequently, in the nonlinear case it is not correct to simply add the analytical crack tip solution to the global solution. This can only be done after these special solutions are modified appropriately. The following development is included to make this issue clear.

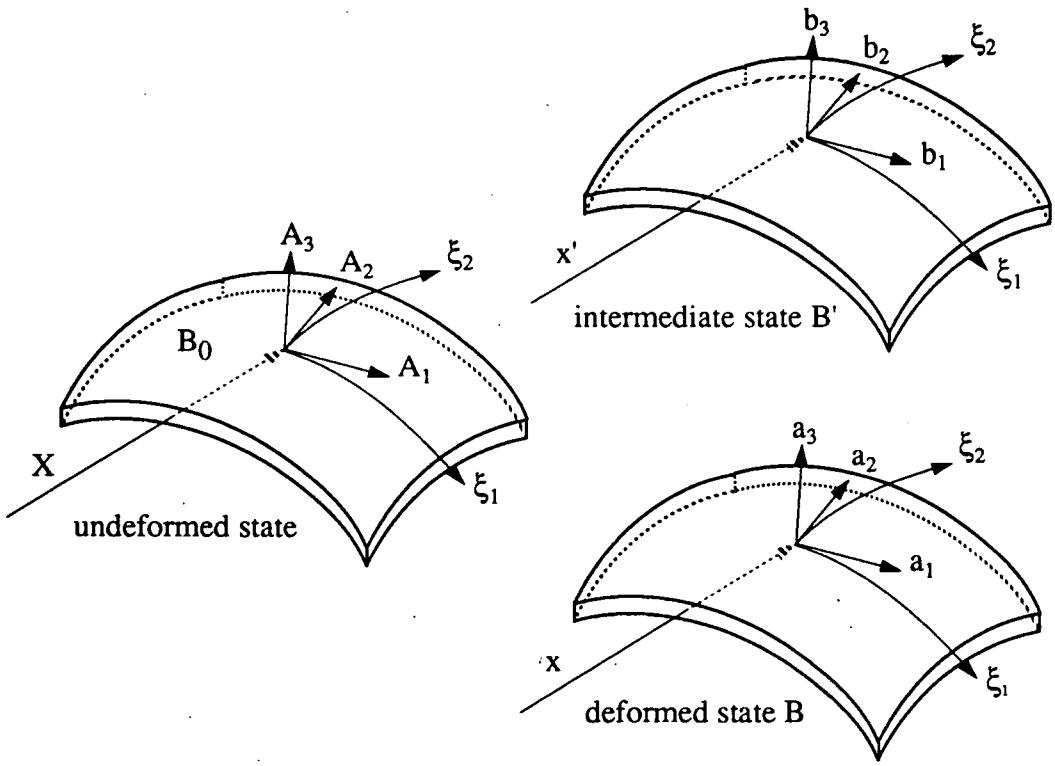


Figure 3: Definition of the undeformed, the intermediate and the final deformed state

Suppose that the shell mid surface in the undeformed state B_0 is described by the mapping (Figure 3)

$$\mathbf{X} = \mathbf{X}(\xi) = \mathbf{y}(\xi, 0) \tag{2.4}$$

After deformation, in state B, the shell mid surface is described by

$$\mathbf{x} = \mathbf{x}(\xi, \zeta) = \mathbf{y}(\xi, \zeta) \tag{2.5}$$

where ζ is the parameter that represents the intensity of the deformation or the time, whichever is appropriate. The inclusion of the evolution parameter ζ in this discussion is purely for the sake of clarity of exposition.

The difference between the two states B_0 and B is the displacement field

$$\underline{\mathbf{u}} = \underline{\mathbf{u}}(\xi, \zeta) = \mathbf{x}(\xi, \zeta) - \mathbf{X}(\xi) \tag{2.6}$$

The displacement $\underline{\mathbf{u}}$ at the crack tip comprises the global field $\underline{\mathbf{u}}_1$ furnished by the finite element discretization

$$\underline{\mathbf{u}}_1 = \underset{\sim}{\mathbf{U}}_1(\xi, \zeta) = \mathbf{N}^T(\xi)\,\underline{\mathbf{q}}(\zeta) \tag{2.7}$$

and the singular crack tip solution

$$\underline{\mathbf{u}}_2 = \underset{\sim}{\mathbf{U}}_2(\xi, \zeta) = \Theta(\xi)\,\mathbf{k}(\zeta) \tag{2.8}$$

While the global field $\underline{\mathbf{u}}_1$ extends itself over the whole domain, the singular field $\underline{\mathbf{u}}_2$ is nonzero only in a small domain around the crack tip.
The singular crack tip shape functions $\underline{\mathbf{u}}_2$ are constructed of the classical crack tip solutions derived in the linear theory of elasticity. This solution is not directly applicable to the nonlinear case because we have to take into account the rotation of the crack tip in space during the deformation process. If the shape functions corresponding to the classical crack tip solutions are given by $\mathbf{v}(\xi)$, the function $\underline{\mathbf{u}}_2$ is given by

$$\underline{\mathbf{u}}_2 = \mathbf{R}(\xi_T)\,\mathbf{v}(\xi, \zeta) = \mathbf{k}(\zeta)\,\mathbf{R}(\xi_T)\,\mathbf{v}_0(\xi) \tag{2.9}$$

where $\mathbf{R}(\xi_T)$ denotes the rotation at the crack tip (ξ_T = the location of the crack tip). Note that the rotation matrix is dependent on the solution $\underline{\mathbf{u}}_1(\zeta)$ so that one could write

$$\underline{\mathbf{u}}_2 = \mathbf{R}(\xi_T, \zeta)\,\mathbf{v}(\xi, \zeta) = \mathbf{k}(\zeta)\,\mathbf{R}(\xi_T, \zeta)\,\mathbf{v}_0(\xi) \tag{2.10}$$

The configuration in the deformed state can be seen to be attained in two stages, the first stage by the displacement $\underline{\mathbf{u}}_1$, the second by the addition of $\underline{\mathbf{u}}_2$. In this way we identify an intermediate state B' described by

$$\mathbf{x}' = \mathbf{X} + \underline{\mathbf{u}}_1 = \mathbf{y}(\xi, 0) + \underset{\sim}{\mathbf{U}}_1(\xi, \zeta) \tag{2.11}$$

The base vectors that belong to the undeformed state are given by

$$\mathbf{A}_i = \frac{\partial}{\partial \xi^i}\mathbf{y}(\xi, 0) \tag{2.12}$$

In the intermediate state the base vectors are

$$\mathbf{b}_i = \frac{\partial}{\partial \xi^i}\mathbf{x}'(\xi,\zeta) = \mathbf{A}_i + \frac{\partial}{\partial \xi^i}(\underset{\sim}{U}_1(\xi,\zeta)) = \mathbf{A}_i + \underset{\sim}{u}_{1,i} \tag{2.13}$$

Note that a comma followed by an index (i) denotes partial differentiation with respect to ξ_i. In our formulation we use the Green-Lagrange strains, which can be written as

$$G_{ij} = \frac{1}{2}\{(\mathbf{A}_i)^T\mathbf{u}_{,j} + (\mathbf{u}_{,i})^T\mathbf{A}_j + (\mathbf{u}_{,i})^T\mathbf{u}_{,j}\} \tag{2.14}$$

Substitution of the composite field $\underset{\sim}{u}_1 + \underset{\sim}{u}_2$ gives

$$\begin{aligned} G_{ij}(\underset{\sim}{u}_1, \mathbf{Rv}) = \frac{1}{2}\{&(\mathbf{A}_i)^T\underset{\sim}{u}_{1,j} + (\underset{\sim}{u}_{1,i})^T\mathbf{A}_j + (\underset{\sim}{u}_{1,i})^T\underset{\sim}{u}_{1,j} \\ &+ (\mathbf{A}_i + \underset{\sim}{u}_{1,i})^T\mathbf{Rv}_{,j} + (\mathbf{Rv}_{,i})^T(\mathbf{A}_j + \underset{\sim}{u}_{1,j}) + (\mathbf{Rv}_{,i})^T\mathbf{Rv}_{,j}\} \end{aligned} \tag{2.15}$$

or

$$\begin{aligned} G_{ij}(\underset{\sim}{u}_1, \mathbf{Rv}) = G_{ij}(\underset{\sim}{u}_1) + \frac{1}{2}\{&(\mathbf{A}_i + \underset{\sim}{u}_{1,i})^T\mathbf{Rv}_{,j} + (\mathbf{Rv}_{,i})^T(\mathbf{A}_j + \underset{\sim}{u}_{1,j}) \\ &+ (\mathbf{v}_{,i})^T\mathbf{R}^T\mathbf{Rv}_{,j}\} \end{aligned} \tag{2.16}$$

Substituting equation (2.13) into (2.16) and making use of the orthogonality of the rotation matrix $\mathbf{R}$ (thus $\mathbf{R}^T\mathbf{R} = \mathbf{I}_3$) yields

$$G_{ij}(\underset{\sim}{u}_1, \mathbf{Rv}) = G_{ij}(\underset{\sim}{u}_1) + \frac{1}{2}\{(\mathbf{b}_i)^T\mathbf{Rv}_{,j} + (\mathbf{Rv}_{,i})^T\mathbf{b}_j + (\mathbf{v}_{,i})^T\mathbf{v}_{,j}\} \tag{2.17}$$

Equation (2.17) can be rewritten as

$$G_{ij}(\underset{\sim}{u}_1, \mathbf{Rv}) = G_{ij}(\underset{\sim}{u}_1) + G^*_{ij}(\underset{\sim}{u}_1, \mathbf{Rv}) \tag{2.18}$$

where

$$G^*_{ij}(\underset{\sim}{u}_1, \mathbf{Rv}) = \frac{1}{2}\{(\mathbf{b}_i)^T\mathbf{Rv}_{,j} + (\mathbf{Rv}_{,i})^T\mathbf{b}_j + (\mathbf{v}_{,i})^T\mathbf{v}_{,j}\} \tag{2.19}$$

G^*_{ij} describes the increase of the deformation in going from B' to B. It follows from the foregoing that G^*_{ij} is only locally defined, i.e. in a small area Ω^* around the crack tip. Everywhere outside this area G^*_{ij} is equal to zero. As long as the initial curvature and the deformation $G_{ij}(\underline{u}_1)$ remain small in Ω^*, which is a reasonable assumption, the variation of $\mathbf{b}_i$ over this area will also be small. Therefore, we can replace the base vectors $\mathbf{b}_i(\xi)$ belonging to state B' in equation (2.19) by base vectors $\mathbf{b}_i(\xi_T) = \mathbf{b}_{i(cr)}$ that are attached to a point determined by ξ_T in the region Ω^* (for which the node at the crack tip presents itself as a natural candidate).
Let **v** be specified by

$$\mathbf{v} = \mathbf{v}_k(\xi)\,\mathbf{A}(\xi_T)_k = \mathbf{v}_k(\xi)\,\mathbf{A}_{k(cr)} \tag{2.20}$$

where we assume that the base vectors $\mathbf{A}_{k(cr)}$ at the crack tip for the undeformed state are unit vectors and mutually orthogonal. Substitution in equation (2.19) yields

$$G^*_{ij}(\underline{u}_1, \mathbf{Rv}) = \frac{1}{2}\{(\mathbf{b}_{i(cr)})^T\mathbf{R}\mathbf{A}_{k(cr)}\mathbf{v}_{k,j} + \mathbf{v}_{k,i}(\mathbf{R}\mathbf{A}_{k(cr)})^T\mathbf{b}_{j(cr)} + \mathbf{v}_{k,i}\mathbf{v}_{k,j}\} \tag{2.21}$$

The base vectors $\mathbf{b}_{i(cr)}$ are obtained from the base vectors $\mathbf{A}_{k(cr)}$ by a rotation, which we identify with **R**, and a small deformation caused by the field $\underline{u}_1$. It is now natural and consistent with our previous assumptions to neglect the deformation of the triad $\mathbf{A}_{k(cr)}$ and replace $\mathbf{b}_{i(cr)}$ by $\mathbf{R}\mathbf{A}_{k(cr)}$. The additional strains G^*_{ij} that arise can then be simplified to the expressions

$$G^*_{ij}(\underline{u}_1, \underline{u}_2) = G^*_{ij}(\underline{u}_1, \mathbf{Rv}) = \frac{1}{2}(\mathbf{v}_{i,j} + \mathbf{v}_{j,i} + \mathbf{v}_{k,i}\mathbf{v}_{k,j}) \tag{2.22}$$

Comparing this additional Green-Lagrange strain G^*_{ij} with the (truncated) strain field derived on the basis of linearized elasticity theory one can see that the only difference lies in the nonlinear term $\mathbf{v}_{k,i}\mathbf{v}_{k,j}$. The effect of this nonlinear term on the total additional strain G^*_{ij} remains small as long as the displacement gradients are small. However when approaching the crack tip these displacement gradients will increase and so will the nonlinear term $\mathbf{v}_{k,i}\mathbf{v}_{k,j}$. This increase of $\mathbf{v}_{k,i}\mathbf{v}_{k,j}$ will cause a discrepancy between the truncated strain field and the Green-Lagrange strain field.
The concept of singular stress and strain field around the crack tip is derived on the basis of linearized elasticity theory which is formulated in terms of small displacement gradients. Actually this presents a contradiction in terms because the stresses and strains will go to

infinity when closing in to the crack tip. This contradiction is generally accepted as a modeling error that occurs only in a very small neighborhood of the crack tip. However, to be consistent with the singular stress and strain field for the linearized theory the additional Green-Lagrange strain G^*_{ij} also has to be truncated.

In order to linearize the in-plane strains for the rotated orientation of the crack tip zone the rotation matrix has to be determined first. This rotation matrix can be calculated by using the orthogonal triads in the crack tip node for the undeformed and for the rotated crack tip zone. By applying this rotation matrix to the analytical crack tip strains the in-plane strains corresponding to the rotated crack tip zone can be determined. From these modified strains the quadratic component in equation (2.22) can be calculated and subtracted from the total strains thus correcting the in-plane strains as discussed above.

The (truncated) Green-Lagrange strain G^*_{ij} that has to be added to the finite element strains therefore is

$$G^*_{ij}(\underline{u}_1, \underline{u}_2) = \frac{1}{2}(v_{i,j} + v_{j,i}) \tag{2.23}$$

The strain increments caused by insertion of the classical solutions for the crack tip behavior in the global representation of the displacements are thus obtained by a direct evaluation of the Green-Lagrange functional expressions in terms of the local components of this displacement field. It is further noted that the two fields $\underline{u}_1$ and $\underline{u}_2$ are not coupled in the strain measures, but only in the specific elastic energy $E = 0.5\,(G_{ij}\,C_{ijkl}\,G_{kl})$, at least to an acceptable degree of approximation.

2.1.1.4 The resulting nonlinear equations

The nonlinear equations that result from the foregoing definitions can be written as

$$\mathbf{F}(\underline{q}, \mathbf{k}, \lambda) = 0 \tag{2.24}$$

where λ represents the intensity of the loading, $\mathbf{k}$ is the vector of stress intensity factors and $\underline{q}$ is the vector of total nodal displacements.

These total nodal displacements for the enriched elements are obtained by summation of the analytical displacement solution corresponding to the stress intensity factors and the underlying conventional finite element displacement field (Figure 4). Thus

$$\underline{q}_i = \underline{q}_i + \mathbf{k}\mathbf{R}\underline{v}_i \tag{2.25a}$$

or

$$\underset{\sim}{q}_i = \underset{\sim}{q}_i - \mathbf{k}\mathbf{R}\underline{v}_i \tag{2.25b}$$

where

- $\underline{v}_i$ are the nodal values of the analytical displacement solution corresponding to the stress intensity factors,
- $\underset{\sim}{q}_i$ are the nodal displacements determined by the underlying finite element displacement formulation,
- $\underset{\sim}{q}_i$ are the total nodal displacements,
- $\mathbf{R}$ represents the rotation of the crack tip with respect to the undeformed state and
- $\mathbf{k}$ is the vector of stress intensity factors.

The total displacement field can be written now as

$$\begin{aligned} \underset{\sim}{u} &= \mathbf{N}^T(\xi)\,\underset{\sim}{q}_i + \mathbf{k}\mathbf{R}\mathbf{v} \\ &= \mathbf{N}^T(\xi)\,\underset{\sim}{q}_i + \mathbf{k}\mathbf{R}\,(\mathbf{v} - \mathbf{N}^T(\xi)\,\underline{v}_i\,) \end{aligned} \tag{2.26}$$

where $\mathbf{N}^T(\xi)$ are the finite element shape functions.
Returning to equation (2.24) we decompose this set of nonlinear equations in the form

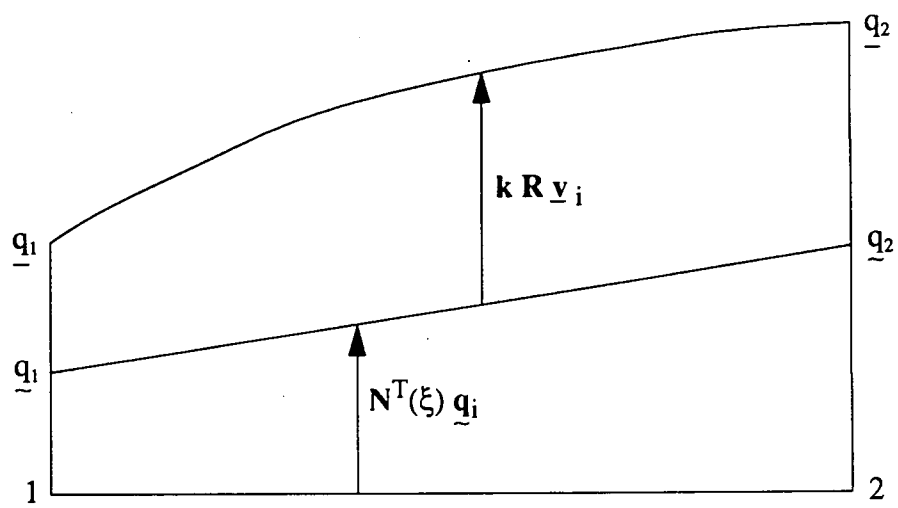


Figure 4: Separation of the analytical displacement solution corresponding to the stress intensity factors and the underlying finite element displacement field

$$\mathbf{f}(\underline{q}_i, \lambda) + \mathbf{f}^*(\underline{q}_i, \mathbf{k}, \lambda) = 0 \qquad (2.27a)$$

$$\mathbf{h}^*(\mathbf{k}, \lambda) = 0 \qquad (2.27b)$$

where

$$\mathbf{f}, \mathbf{f}^* \in \mathbf{R}_N \quad ; \quad \mathbf{h}^* \in \mathbf{R}_k$$

The first term in equation (2.27a) corresponds to the contributions of the finite element discretization scheme. The two other terms in these equations arise when we apply the enrichment scheme described in the previous sections. Equations (2.27a-b) can be seen to be derived from the requirement that the total potential energy of the structure with its loads must be stationary. The contributions $\mathbf{f}$, $\mathbf{f}^*$ and $\mathbf{h}^*$ are then computed from

$$\mathbf{f} = \frac{\partial P}{\partial \underline{q}_i} \quad ; \quad \mathbf{f}^* = \frac{\partial P^*}{\partial \underline{q}_i} \quad ; \quad \mathbf{h}^* = \frac{\partial P^*}{\partial \mathbf{k}} \qquad (2.28)$$

where the potential energy is given in two parts.

$$P_{total} = P(\underline{q}_i, \lambda) + P^*(\underline{q}_i, \mathbf{k}, \lambda) \qquad (2.29)$$

The first term in equation (2.29) refers to the global part of the formulation while the second term P^* is due to the mode enrichment process.

2.1.1.5 Solution of the corrector equations

The nonlinear equations (2.27a-b) are solved for a range of values of the loading depending on the particular requirements for the analysis of the problem at hand. This solution process is carried out step by step by variation of the load parameter λ or, more generally, by variation of a generalized path parameter η. It is characteristic for these type of procedures that each new step along the loading path is carried out by a predictor corrector scheme [21].
The addition of the modes corresponding to the stress intensity factors to the regular discretization scheme however has an impact on the solution algorithm that needs some clarification. The changes can best be described at the level of the corrector equations.
The solution of the corrector equations in each new step of the step by step solution algorithm furnishes sequentially a set of corrections $\Delta\tau^i$: i=1, 2, 3 ... to the predicted solution τ^0 of the

equation set (2.24). This system of equations is usually a linearization of (2.24) and for the sake of simplicity we will assume that this linearization corresponds to Newton's method.
The governing equations on which the operations are carried out are first written as

$$\mathbf{H}(\mathbf{t}, \eta) = 0 \tag{2.30}$$

where

$$\mathbf{t} = \begin{bmatrix} \underline{\mathbf{q}} \\ \mathbf{k} \\ \lambda \end{bmatrix} \in \mathbf{R}_N + \mathbf{R}_k + 1 \qquad \mathbf{H} = \begin{bmatrix} \mathbf{F}(\mathbf{t}) \\ \mathbf{g}(\mathbf{t}, \eta) \end{bmatrix} \tag{2.31}$$

The extra equation that is introduced here is a device to introduce the path parameter η that measures the distance between the solution points along the loading path.
Newton's method is based on the linearization

$$\mathbf{H}_x(\mathbf{t}, \eta)\,\Delta\tau = -\mathbf{H}(\mathbf{t}, \eta) \tag{2.32}$$

which we call the Newton form of equation (2.24). The configuration τ is here some approximation to the actual solution, while $\Delta\tau$ stands for the correction to this approximation at the prescribed value of η that determines a particular point on the loading path.
In matrix form equation (2.32) can be written as

$$\begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} & \mathbf{F}_\lambda \\ \mathbf{S}_{12}^T & \mathbf{S}_{22} & \mathbf{F}_{k\lambda} \\ \mathbf{n}_q & \mathbf{n}_k & n_0 \end{bmatrix} \begin{bmatrix} \Delta\underline{\mathbf{q}} \\ \Delta\mathbf{k} \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f}_i \\ \mathbf{f}_{k\lambda} \\ 0 \end{bmatrix} \tag{2.33}$$

where

- $\mathbf{S}_{11}$ is the global stiffness matrix corresponding to the conventional finite element displacement formulation,
- $\mathbf{S}_{12}$ is the global (sparse) stiffness matrix containing the coupling terms between the conventional displacement formulation and the analytical crack tip displacement solution,
- $\mathbf{S}_{22}$ is the global stiffness matrix containing terms due to the analytical crack tip displacement solutions only,

$\mathbf{F}_\lambda$ is the vector of nodal forces corresponding to the conventional displacement field,
$\mathbf{F}_{k\lambda}$ is the load vector corresponding to the analytical crack tip displacement solutions,
$\mathbf{n}_q$ is the tangential vector to the solution field for the current load,
$\mathbf{n}_k$ is the 'stress intensity'-part of the total tangential vector,
n_0 is a constant that defines the auxiliary surface needed in the continuation procedure,
$\Delta \mathbf{k}$ is the correction of the stress intensity factor(s),
$\Delta \underline{\mathbf{q}}$ is the correction of the nodal displacements,
$\Delta\lambda$ is the correction of the load step size,
$\mathbf{f}_i$ is the vector of residual forces corresponding to the conventional displacement field and
$\mathbf{f}_{k\lambda}$ is the vector of residual forces corresponding to the analytical crack tip displacement solutions.

As already stated above $\mathbf{S_{12}}$ contains the coupling terms between the conventional displacement formulation and the analytical displacement solutions, so

$$\mathbf{S_{12}} = \int_V \delta \mathbf{G}^T_{ij_{conv}} \mathbf{C}_{ijkl} \mathbf{G}_{kl_{anal}} dV \tag{2.34}$$

This also implies that matrix $\mathbf{S_{12}}$ will contain a considerable number of nonzero values because a number of elements around the crack tip are involved in the enrichment process. Entering this sparse matrix together with the conventional global stiffness matrix in an equation solver would be inconvenient because it would destroy the bandwidth. Therefore the corrector equations (2.33) will be solved with a Block-Gauss elimination scheme as follows. First equation (2.33) is written as

$$\mathbf{S_A}\Delta\mathbf{a} + \mathbf{S_B}\Delta\mathbf{b} = \mathbf{r_a} \tag{2.35a}$$

$$\mathbf{S_C}\Delta\mathbf{a} + \mathbf{S_D}\Delta\mathbf{b} = \mathbf{r_b} \tag{2.35b}$$

where

$$\mathbf{S_A} = \begin{bmatrix} \mathbf{S_{11}} \end{bmatrix} \qquad \mathbf{S_B} = \begin{bmatrix} \mathbf{S_{12}}\ \mathbf{F}_\lambda \end{bmatrix}$$

$$\mathbf{S_C} = \begin{bmatrix} \mathbf{S}^T_{12} \\ \mathbf{n}_q \end{bmatrix} \qquad \mathbf{S_D} = \begin{bmatrix} \mathbf{S_{22}}\ \mathbf{F}_{k\lambda} \\ \mathbf{n}_k\ \ n_0 \end{bmatrix}$$

$$\Delta\mathbf{a} = \begin{bmatrix} \Delta\underline{\mathbf{q}} \end{bmatrix} \qquad \Delta\mathbf{b} = \begin{bmatrix} \Delta\mathbf{k} \\ \Delta\lambda \end{bmatrix}$$

$$\mathbf{r_a} = \begin{bmatrix} \mathbf{f_i} \end{bmatrix} \qquad \mathbf{r_b} = \begin{bmatrix} \mathbf{f_{k\lambda}} \\ 0 \end{bmatrix} \tag{2.36}$$

Working out equation (2.35a) yields

$$\Delta\mathbf{a} = \mathbf{S_A^{-1}} \mathbf{r_a} - \mathbf{S_A^{-1}} \mathbf{S_B} \Delta\mathbf{b} \tag{2.37}$$

and substitution of equation (2.37) into equation (2.35b) will result in

$$\mathbf{S_C S_A^{-1}} \mathbf{r_a} - \mathbf{S_C S_A^{-1}} \mathbf{S_B} \Delta\mathbf{b} + \mathbf{S_D} \Delta\mathbf{b} = \mathbf{r_b} \tag{2.38}$$

or

$$(\mathbf{S_D} - \mathbf{S_C S_A^{-1}} \mathbf{S_B})\, \Delta\mathbf{b} = \mathbf{r_b} - \mathbf{S_C S_A^{-1}} \mathbf{r_a} \tag{2.39}$$

From this equation $\Delta\mathbf{b}$ can be calculated and $\Delta\mathbf{a}$ is obtained from equation (2.37) by substitution of $\Delta\mathbf{b}$.
This technique for solving the corrector equations is already applied in the B2000 routine 'b2cont' when dealing with the additional path equation because otherwise this path equation would completely destroy the bandwidth of the matrix. Therefore the solution process discussed in this section actually implies an extension of the method that is already implemented in 'b2cont'.

2.1.1.6 A posteriori mode enrichment

In section (2.1.1.3) it was shown that the difference between the intermediate state B' and the final deformed state B is furnished by the inclusion of the analytical displacement solution $\underline{\mathbf{u}}_2$ in the total displacement field. This analytical displacement solution is only locally defined and therefore its influence on the total displacement solution is mainly restricted to a small area around the crack tip. Actually the enrichment process represents a linear technique applied to a small part of a geometrically nonlinear deformed structure which gives rise only to small changes in the general solution. Therefore one can apply mode enrichment as an additional step after the nonlinear state of deformation of the structure described by the global field $\underline{\mathbf{u}}_1$ is completed. It turns out that because of the linear character of the mode enrichment process the stress intensity factors and thus the final deformed state is obtained within a few iterations only. This technique is particularly useful in cases where the knowledge of the stress

intensity factors is not required at each point of the load-deformation path but only for the final load value.

There is yet another reason why this a posteriori technique is attractive to use. As will be described in section 2.1.1.7 a higher order Gaussian integration scheme has to be applied for the enriched elements to get accurate solutions for the stress intensity factors. This means that the formation of the stiffness matrices for the enriched elements is rather time consuming due to the high number of integration points and the evaluation of the additional stiffness terms in these integration points. A posteriori application of the mode enrichment technique is thus an effective means to reduce the volume of the computation because the higher order integration scheme has to be applied only once (namely after the nonlinear deformation of the structure is completed).

This technique is implemented in the B2000 processor 'b2contcrack' via an interactive command ENRI (see Section 2.2). Furthermore, another technique is implemented where mode enrichment is applied for the complete loading history while the calculation is repeated once for the final load step without enriching the crack tip zone (via the interactive input command REMO). These final results can be used to perform an error calculation because the displacements are now governed by the finite element formulation only. This also has the advantage that, after the solution data is mapped from an 'old' finite element mesh onto a 'new' one with an extended crack, the 'old' crack tip region in the 'new' finite element mesh does not have to be enriched when relaxing the residual loads that keep the new crack increment closed.

2.1.1.7 Numerical integration

In finite element analysis the integrals involved in the equilibrium equations are evaluated numerically. The type and order of this numerical integration procedure has to be chosen in such a way to achieve an acceptable level of accuracy for the finite element results (including the stress intensity factors).

A widely used numerical integration technique is the so-called Gaussian quadrature procedure [1]. In this procedure the position and the weight of the integration points is optimized in such a way that the finite element matrices, containing the polynomial interpolation functions for the isoparametric elements, are integrated exactly. This exact integration can be achieved with a low order (2x2 or 3x3) integration scheme depending on the type of element used (4 or 8/9 nodes).

The enriched elements however also contain shape functions corresponding to the analytical crack tip solutions for the different modes of cracking. Due to these additional shape functions one can not suffice with a low order integration scheme for these elements[2, 8, 11, 12, 19, 25, 26, 28]. In [14,15] a few example calculations are presented and it is shown that a higher order integration scheme of 10x10 will give converged solutions.

2.1.2 The elastic T-term via mode enrichment

2.1.2.1 Introduction

The elastic T-term is calculated by extracting it directly from the finite element displacement field around the crack tip. In the literature one can find several other techniques for calculating the elastic T-term. However, these methods make use of additional load systems [7,13] or weight functions [23] and therefore involve much more effort to determine the elastic T-term, especially when dealing with a geometrically nonlinear problem.
The analytical plane stress displacement solution around the crack tip corresponding to the membrane mode I fracture mode can be written as

$$u = \frac{K_I}{\mu}\sqrt{\frac{r}{2\pi}}\cos\frac{\theta}{2}\left(\frac{1-\nu}{1+\nu}+\sin^2\left(\frac{\theta}{2}\right)\right)+\frac{T}{E}r\cos\theta+$$
$$\sum_{n=3}^{\infty}\frac{H_n}{2\mu}r^{n/2}\left(\frac{3-\nu}{1+\nu}\cos\frac{n\theta}{2}-\frac{n}{2}\cos\left(\frac{n}{2}-2\right)\theta+\left(\frac{n}{2}+(-1)^n\right)\cos\frac{n\theta}{2}\right) \qquad (2.40a)$$

$$v = \frac{K_I}{\mu}\sqrt{\frac{r}{2\pi}}\sin\frac{\theta}{2}\left(\frac{2}{1+\nu}-\cos^2\left(\frac{\theta}{2}\right)\right)-\frac{T}{E}\nu r\sin\theta+$$
$$\sum_{n=3}^{\infty}\frac{H_n}{2\mu}r^{n/2}\left(\frac{3-\nu}{1+\nu}\sin\frac{n\theta}{2}+\frac{n}{2}\sin\left(\frac{n}{2}-2\right)\theta-\left(\frac{n}{2}+(-1)^n\right)\sin\frac{n\theta}{2}\right) \qquad (2.40b)$$

where μ is the shear modulus, ν is Poisson's ratio and E is Young's modulus.
The conventional finite element displacement formulation is not able to describe the first part of the displacement field corresponding to the stress intensity factor. In our method this 'singular' part of the displacement field is taken care of via the mode enrichment technique. In that case the finite element formulation has to approximate the displacements for the higher order terms starting with the elastic T-term.
The displacement field for the elastic T-term varies in a linear way in x- and y-direction and therefore can be described exactly by the finite element displacement field. However, the conventional finite element formulation is not able to describe the analytical displacement fields for the other higher order terms exactly. This poses a difficulty, because the interpolation functions of the finite element mesh try to approximate the lower and higher order terms at the same time. Consequently, the linear part of the finite element displacement field, which should correspond to the elastic T-term, is affected by the higher order terms, thus impairing the accuracy of the calculated elastic T-term. How this difficulty can be circumvented (to some extend) will be discussed in the next section.

2.1.2.2 Solution strategy

According to equation (2.40a) the analytical displacement solution u ahead of the crack tip ($\theta=0^o$) contains, besides the elastic T-term, a contribution not only of the stress intensity factor K_I but also of all the other higher order terms. The linear part of the finite element displacement field in this region will be disturbed significantly by these higher order terms. Consequently it is difficult to compute the elastic T-term from the displacement field ahead of the crack tip with a reasonable accuracy.

On the other hand, behind the crack tip ($\theta=180^o$) the analytical displacement solution only contains contributions from the elastic T-term and from the higher order terms with even index n (H_4, H_6, H_8 etc.). Further there is no contribution of the displacement solution corresponding to K_I. This implies that behind the crack tip the disturbance of the linear displacement field is reduced to a minimum and that the elastic T-term can be computed rather accurately along the crack faces (which are situated at $\theta=180^o$). This accuracy can even be increased by the use of quadratic (8- or 9-node) elements which are able to describe the analytical displacement fields corresponding to T and H_4 exactly.

One of the problems is that the nodal displacements along the crack face (which will be used for calculating the elastic T-term) also have to describe the displacement field within the element they are part of. Such an element covers an area for which $\theta<180^o$. Consequently, the mode I stress intensity factor and all the higher order terms are still able to affect this linear displacement field along the crack face to some extend.

It is obvious that the major part of this disturbance is due to the displacement field corresponding to K_I, because this field is dominant around the crack tip. In our approach this problem is solved by enriching the crack tip elements. However, there still is a contribution of the higher order terms with even index n (H_4, H_6, H_8 etc.) to the total displacement field along the crack faces. This means that for a general case the stress along the crack face is not constant. Therefore it was decided to use the following technique for calculating the elastic T-term.

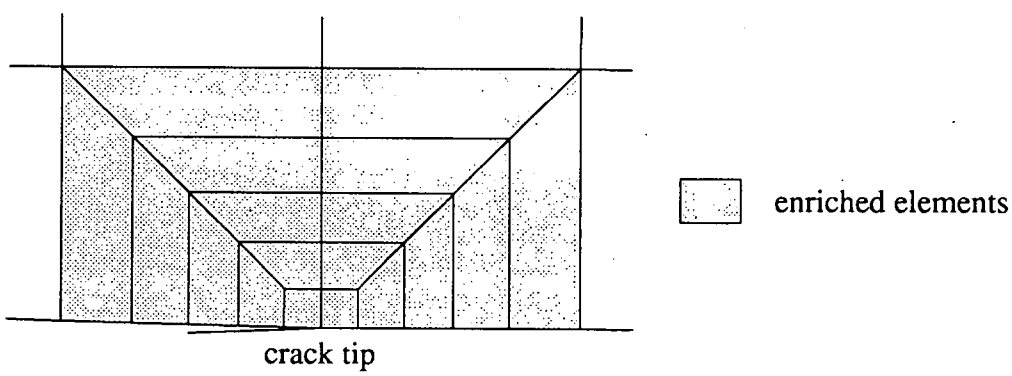


Figure 5: Refined finite element mesh around the crack tip

First of all the finite element mesh around the crack tip is refined by introducing a number of new element rings inside the original crack tip elements (Figure 5). Next, the mode enrichment technique is applied to all the elements in this crack tip zone where the outer ring of elements serves as transition zone between the enriched and the conventional elements. The enriched and transition elements are integrated with a higher order 10x10 Gauss integration scheme (see section 2.1.1.7) while the conventional elements are integrated with a low order 2x2 or 3x3 integration scheme.

After completing the finite element calculation the element stresses corresponding to the linear finite element displacement field tangential to the crack face are determined along the element sides that coincide with this crack face. The final step is then to determine the elastic T-term from these element stresses. This will be accomplished by making a least square fit over a number of element sides behind the crack tip. The details of this technique will be described in the next section.

2.1.2.3 Numerical procedure

The finite element displacement solution for an element side (of a quadratic quadrilateral element) that coincides with the crack face can be written as (Figure 6)

$$u = -\frac{1}{2}\xi_1 (1-\xi_1)\, u_1 + (1-\xi_1^2)\, u_3 + \frac{1}{2}\xi_1 (1+\xi_1)\, u_2 \qquad (2.41)$$

where ξ_1 is one of the two parameters that are used to define the isoparametric quadrilateral element ($\xi_1 \in [-1, 1]$).

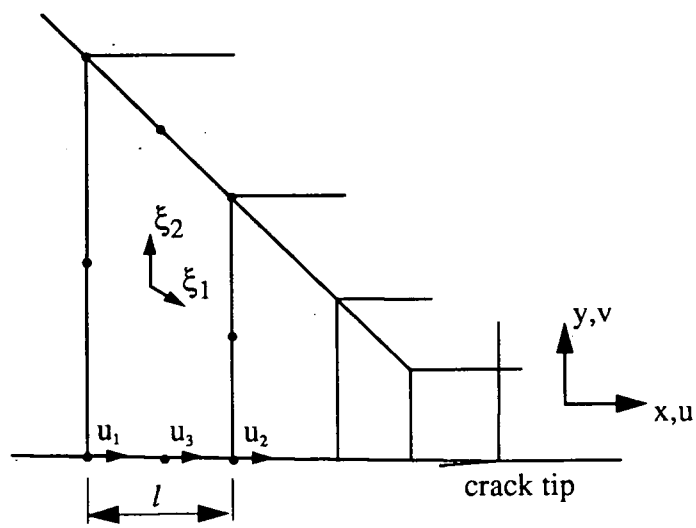


Figure 6: Part of the refined finite element mesh behind the crack tip

Because the elastic T-term represents a constant stress parallel to the crack faces we will only use the 'constant stress part' to calculate this elastic T-term. Therefore we are interested only in the linear part of the finite element displacement field along these crack faces, thus

$$u_{lin} = \frac{1}{2}\xi_1 (u_2 - u_1) \tag{2.42}$$

Differentiating with respect to x yields

$$\varepsilon_{lin} = \frac{\partial u}{\partial x} = \frac{\partial u}{\partial \xi_1}\frac{\partial \xi_1}{\partial x} = \frac{1}{2}\frac{\partial \xi_1}{\partial x}(u_2 - u_1) = \frac{1}{l}(u_2 - u_1) \tag{2.43}$$

For geometrically nonlinear problems one has to note that the crack tip region will rotate in space during deformation. Therefore the in plane displacement field after rotation of the crack tip has to be considered for the calculation of the elastic T-term. For the strain tangential to the crack face this yields (Figure 7)

$$\tilde{\varepsilon}_{lin} = \frac{\partial \tilde{u}}{\partial \tilde{x}} = \frac{\partial u}{\partial x} + \frac{1}{2}\left(\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial w}{\partial x}\right)^2\right) \tag{2.44}$$

or

$$\tilde{\varepsilon}_{lin} = \frac{1}{l}(u_2 - u_1) + \frac{1}{2l^2}\left((u_2 - u_1)^2 + (v_2 - v_1)^2 + (w_2 - w_1)^2\right) \tag{2.45}$$

The element stresses corresponding to the elastic T-term can be obtained by simply multiplying these strains with the elasticity constants (see equation (2.40a)).
Now, in order to establish a least square fit over these element stresses, it is first necessary to choose a continuous function for the stress field behind the crack tip. As already mentioned in the previous section, this stress field is governed by the elastic T-term and the higher order terms with even index n. It follows from the previous remarks that this continuous stress function should look as follows:

$$\sigma(x) = C_0 + C_1 x_c + C_2 x_c^2 + C_3 x_c^3 + \ldots\ldots \tag{2.46}$$

In this equation C_0 represents the elastic T-term while C_1, C_2 etc. correspond to the higher order terms. Further x_c denotes the distance to the crack tip.

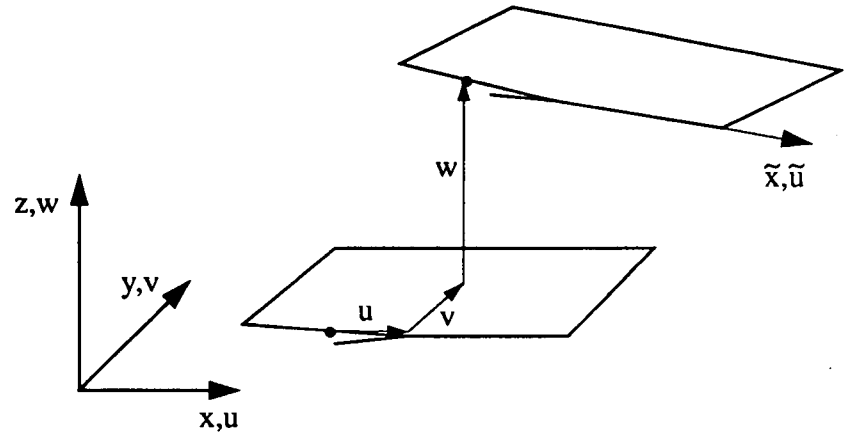


Figure 7: Crack tip before and after rotation

According to the least square fit technique the error between this continuous function for the stress field and the discrete stresses calculated in the elements behind the crack tip has to be minimized where this error is defined as

$$E = \sum_{i=1}^{n_e} (\sigma(x_i) - \sigma_i)^2 \qquad (2.47)$$

In this equation x_i denotes the distance of the midpoint of the element sides along the crack face to the crack tip while $\sigma(x_i)$ and σ_i are the values of the continuous stress field $\sigma(x_c)$ and the discrete finite element stress at point x_i respectively (Figure 8). Further n_e represents the number of elements over which this least square fit is performed.
Minimizing the error E error with respect to the coefficients C_n in the polynomial function yields

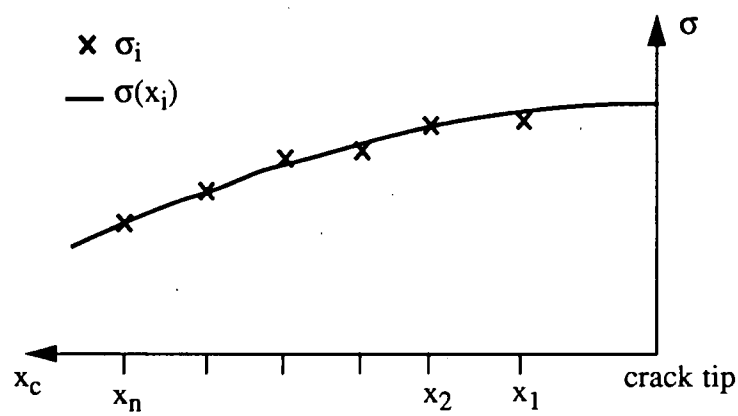


Figure 8: Extrapolation of the element stresses to the crack tip via the least square fit procedure

$$\frac{\partial E}{\partial C_n} = 0 \tag{2.48}$$

for all coefficients C_n. This results in a matrix equation the order of which is determined by the number of coefficients in the continuous stress function σ. Finally the elastic T-term (C_0 in equation (2.46)) is obtained by solving this matrix equation.
The question remains how to choose the number of terms that have to be considered in the stress function behind the crack tip (equation (2.46)) in order to render the elastic T-term with a reasonable accuracy. Moreover, it must be determined over which elements behind the crack tip this least square fit must be established.
The answer to these two questions is given in [15]. Summarizing one can say that for most cases one can do with a polynomial order of 2 for equation (2.46) thus

$$\sigma(x) = C_0 + C_1 x_c + C_2 x_c^2 \tag{2.49}$$

Furthermore the crack tip elements immediately surrounding the crack tip as well as the transition elements along the crack faces were shown to have a negative influence on the accuracy of the calculated elastic T-term. Therefore the corresponding element sides are skipped from the least square procedure. In 'b2contcrack' this can be accomplished by defining the interval LSQ1-LSQ2 of nodes over which the least square procedure is performed (see section 2.3.1, input card 23 and Figure 13)

2.2 Interactive input commands

The new processor 'b2contcrack' is an extension of the existing processor 'b2cont' [4] and therefore can deal with all the interactive commands that are defined for this processor. However, for our purpose some special commands have been added concerning the way in which mode enrichment is applied, the final load value, the number of iterations for each load step, the use of the path following technique, imperfections and information to denote whether the calculation is restarted after applying the solution data mapping routine.

Command:	*Description:*
MENRI	Mode enrichment is applied in the crack tip elements and the stress intensity factor(s) and eventually the elastic T-term are calculated for at least 1 load step.

ENRI

Mode enrichment is applied only for the final load step. This way of working can be used if one is only interested in the final values for the stress intensity factor(s) and elastic T-term. It may cause a considerable reduction of the CPU-time as compared to the application of mode enrichment for all load steps, because a lower order integration scheme is applied throughout a substantial part of the nonlinear calculation (see section 2.1.1.6). This command can only be used in combination with MENRI.

REMO

Mode enrichment is applied during the whole loading history. However, for the maximum load value an additional calculation is performed without applying the mode enrichment technique. This additional information can be used to perform an error calculation (chapter 3) or to simplify the opening process of the new crack increment. It removes the necessity of enriching the 'old' crack tip region in the 'new' finite element mesh as the crack extends. This command can only be used in combination with MENRI.

LMAX

The nonlinear calculation will stop precisely at the maximum load value denoted as PAMAX. This means that one does not have to interpolate between the two final equilibrium states (below and above PAMAX) in order to obtain the solution data for PAMAX.

ITMAX

For each load step the maximum number of iterations, defined in the B2000 input file as MAXIT, is applied. This means that, although the convergence criteria are satisfied for a certain problem, the maximum number of iterations will always be completed. Of course, this is not the case if the solutions starts to diverge and a steps size cut or a refactoring step for the stiffness matrix has to be performed. This option is inspired on the fact that the bending stress intensity factors K_{BI} and K_{BII} and the tearing stress intensity factor K_{III} converge in a much slower way than the two membrane stress intensity factors K_I and K_{II}. However, by using the maximum number of iterations, the final corrections to the solutions for K_{BI}, K_{BII} and K_{III} are very small and convergence will be obtained.

ILOADF

The load value λ is fixed during each load step. This option actually implies that the (modified) Newton method is used instead of the arc-length technique. When opening the new crack increment some problems occurred when using the arc-length technique, because the corrections to the load value became so large that it took a considerable number of load steps to reach the final equilibrium state. However, by fixing the load value it turned out that the final equilibrium state could be obtained within 1 or 2 load steps. This option is often used in combination with the option ITMAX.

NOTP

An imperfection is introduced in order to avoid the existence of bifurcation points. This option is used to pass bifurcation points in a smooth way by using the continuation processor. In the first load step the total applied external load is

$$F_{(1)} = d\lambda_{1_{(1)}} F_1 + \lambda_2 F_2$$

while for the rest of the loading history the external load is denoted as

$$F_{(i)} = (\lambda_{1_{(i-1)}} + d\lambda_{1_{(i)}}) F_1 + \lambda_2 F_2$$

The subscripts without brackets denote the load case number while the subscripts within brackets represent the load step number.

Attention : When using this option the B2000 input file for 'b2contcrack' has to be extended as follows:

LCB = 2	load case 2 added
PBS = λ_2	load value for the imperfection load
DPBS = 0	PBS may not increase during the loading history
PBMAX = 0	
FORCES: case 2	imperfection forces have to be defined as 'case 2'

NMESH

The nonlinear calculation is proceeded again after mapping the solution data from the 'old' mesh onto the 'new' one. This option implies that the equilibrium equations start from a deformed state, where this state does not represent an equilibrium state for the 'new' finite element model. The succeeding steps involved when using this option are as follows:

step 1: The first variations FVAR.GLOB.0 for the mapped displacements and rotations are calculated. These displacements and rotations are situated in the global displacement vector DISP.GLOB.0. Furthermore, the original and rotated orthogonal triads in the element nodes are loaded from database file TRI0.iel and TRIA.iel respectively. These database files (DISP.GLOB.0, TRI0.iel and TRIA.iel) are all produced by the displacement mapping routine (Chapter 4) .

step 2: The 'new' external load vector is calculated as

$$F_e = \lambda_2 F_2 - \text{FVAR.GLOB.0}$$

and written to database file FRCA.GLOB

step 3: The final equilibrium state is calculated now by removing this external force via a load relaxation technique.

Attention : This option has to be used in combination with the option LMAX in order to exactly remove the residual force vector FRCA.GLOB. Furthermore, the B2000 input file for 'b2contcrack' has to be extended as follows:

LCB = 2	load case 2 added
PAS = 1.0	Initial load value for the external load FRCA.GLOB (see above)
DPAS = $d\lambda_1$	initial step for load case 1 used in the load relaxation procedure (-1.0 ≤ DPAS < 0)

PAMAX = 0.	the external forces in vector FRCA.GLOB have to be released
PBS = λ_2	final load value for load case 1 for the nonlinear analysis with the 'old' mesh
DPBS = 0	PBS may not be increased during the new load history
PBMAX = λ_2	= PBS
FORCES	the external forces that were originally defined in load case 1 now have to be defined as load case 2

2.3 Additional in- and output files

2.3.1 Input file 'mode_enrich'

In order to apply the mode enrichment technique the 'b2contcrack' processor needs some special information concerning the enriched element zone. As an additional feature one has the opportunity to use a cylindrical coordinate system in which to define the external loads and boundary conditions. Of course, this implies that the nonlinear analysis is also performed with respect to this cylindrical coordinate system.
In what follows the separate input cards are described in a brief way. For an explanation of the use of some of these cards the reader is referred to the end of this section

card	*description*	*format*
1	ICRSYS : used coordinate system ICRSYS = 1 : cartesian coordinate system ICRSYS = 2 : cylindrical coordinate system	I5
2	CYLC(1:3,1) : cylinder axis (only used if ICRSYS = 2)	3*E12.5
3	CYLC(1:3,2) : point on the cylinder axis (only used if ICRSYS = 2)	3*E12.5
4	IFIXV1,IFIXV2 : initial in plane vectors for the definition of the rotations are either defined on the next input card or determined in the element	2*I5

	processor[*1] (only used if ICRSYS = 2) IFIXV1 = 1 : rotation vector 1 is defined on the next input card IFIXV1 = 0 : rotation vector 1 is determined in the element processor (the same holds for vector 2)	
5	V1FIX : fixed direction for the first in plane rotation vector[*1] (only used if IFIXV1=1 and ICRSYS = 2)	3*E12.5
6	V2FIX : fixed direction for the second in plane rotation vector[*1](only used if IFIXV2=1 and ICRSYS = 2)	3*E12.5
7	NFIG : total number of element configurations[*2] (only used if ICRSYS = 2).	I5
8	ITRTRI(NFIG-1,1:3) : transformation vector denoting how the orthogonal triads determined in the cylindrical coordinate system[*1] have to be transformed in order to obtain the correct orthogonal triads for structural members for which the normal vectors are not aligned with the radial direction[*2] (only used if ICRSYS = 2).	NFIG-1 times 3*I5
9	ICRCKT : identification number of the crack tip node	I5
10	NK1,NK2,NK3,NB1,NB2,NT NK1 = 0: K_I is not calculated NK1 = 1: K_I is calculated (the same holds for NK2 (K_{II}), NK3 (K_{III}), NB1 (K_{BI}), NB2 (K_{BII}) and NT (elastic T-term) Attention: If NB1 = 1 and/or NB2 = 1 and/or NK3 = 1 and/or NT = 1 then also NK1 = 1 and NK2 = 1 (only used if MENRI = 1)	6*I5
11	CRADV(1:3) : in-plane vector at the crack tip node tangential to the crack faces pointing in the direction of the crack [*3] (only used if MENRI = 1)	3*E12.5
12	CRADP(1:3) : in-plane vector at the crack tip node perpendicular to vector CRADV(1:3) [*3] (only used if MENRI = 1)	3*E12.5
13	NCRITM,INTM : number NCRITM of enriched elements used for the calculation of the membrane	2*I5

	stress intensity factors and the corresponding integration order INTM *4(only used if MENRI = 1)	
14	IELENM(1:NCRITM) : identification numbers of the enriched elements used for the calculation of the membrane stress intensity factors *4 (only used if MENRI = 1)	NCRITM*I5
15	NCRITBT,INTBT : number NCRITBT of enriched elements used for the calculation of the bending and tearing stress intensity factors and the corresponding integration order INTBT *4(only used if MENRI = 1 and (NB1 =1 or NB2 = 1 or NK3 = 1))	2*I5
16	IELENBT(1:NCRITBT) : identification numbers of the enriched elements used for the calculation of the bending and tearing stress intensity factors *4 (only used if MENRI = 1 and (NB1 =1 or NB2 = 1 or NK3 = 1)) Attention: Elements that are enriched for the calculation of the bending or tearing stress intensity factors or the elastic T-term also have to be enriched for calculating the membrane stress intensity factors (see input card 10).	NCRITBT*I5
17	NPBRDM : number of **corner** nodes bordering the enriched element domain used for the calculation of the membrane stress intensity factors *4 (only used if MENRI = 1)	I5
18	IPBRDM(1:NPBRDM) : identification numbers of the nodes bordering the enriched element domain that is used for the calculation of the membrane stress intensity factors *4 (only used if MENRI = 1)	NPBRDM*I5
19	NPBRDBT : number of **corner** nodes bordering the enriched element domain used for the calculation of the bending and tearing stress intensity factors *4 (only used if MENRI = 1 and (NB1 =1 or NB2 = 1 or NK3 = 1))	I5
20	IPBRDBT(1:NPBRDBT) : identification numbers of the nodes bordering the enriched element domain that is used for the calculation of the bending and tearing stress intensity factors *4 (only used	NPBRDBT*I5

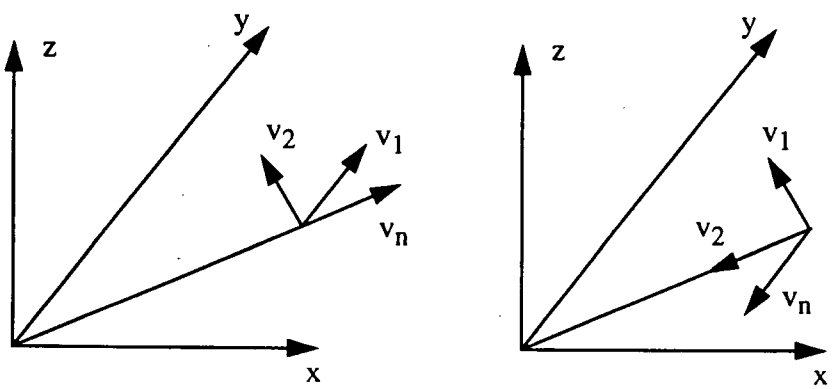
	if MENRI = 1 and (NB1 =1 or NB2 = 1 or NK3 = 1))	
21	IEROT(1:2) : identification number of the 2 elements behind the crack tip that are used for linearizing the in-plane strains[*5] (only used if MENRI = 1)	2I5
22	IPROT(1:3) : identification number of the3 nodes behind the crack tip that are used for linearizing the in-plane strains[*5] (only used if MENRI = 1)	2I5
23	NUMNOD : number of nodes behind the crack tip along the crack faces used to calculate the elastic T-term [*6]. LSQ1-LSQ2 : interval of nodes behind the crack tip used in the least square procedure [*6]. IPOL : polynomial order of the stress function used in the least square procedure [*6]. (used only if MENRI = 1 and NT = 1)	4*I5
24	NODEUP(1:NUMNOD) : identification numbers of the nodes along the upper crack face that are used for the calculation of the elastic T-term [*6].	NUMNOD*I5
25	IELEUP(1:NUMNOD) : identification numbers of the elements along the upper crack face that are used for the calculation of the elastic T-term [*6].	NUMNOD*I5
26	NODELW(1:NUMNOD) : identification numbers of the nodes along the lower crack face that are used for the calculation of the elastic T-term [*6].	NUMNOD*I5
27	IELELW(1:NUMNOD) : identification numbers of the elements along the lower crack face that are used for the calculation of the elastic T-term [*6].	NUMNOD*I5

[*1] The orthogonal triads in the element nodes are calculated in subroutine 'cornrd'. First the normal vector is calculated. The next step is to calculate 1 of the 2 in-plane rotation vectors by determining and normalizing the vector product between this normal vector and 1 of the cartesian unit vectors (in 'cornrd' e_2 (0., 1., 0.) is used for this purpose). However, with this technique one can end up with in-plane rotation vectors which are not aligned with the model boundaries and this can pose some difficulties if one wants to apply boundary conditions along these boundaries. However, the options IFIXV1=1 or IFIXV2=1 offer the opportunity to fix 1 of the 2 in-plane vectors on beforehand. For a cylindrical coordinate system, for instance, one can force 1 of the 2 in-plane rotation vectors to be aligned with the axis of this cylindrical coordinate system. Because the

normal vector in a cylindrical coordinate system is automatically forced to point in the outward radial direction, the second in-plane rotation vector will be aligned with the circumferential direction of the model and this facilitates the application of boundary conditions.

2 If one has to deal with more than 1 structural member, for instance, a shell with rings and stringers, one has to define these different structural members as separate element configurations. For these elements the orthogonal triads, which are determined as described under point 1 have to be modified in order to align them with the element surface. This technique works as follows. First the nodal triads are defined as in point *1. However, the rotation vectors calculated in this way are not situated in the plane of the considered elements while the radial direction is not perpendicular to these elements. For this purpose the orthogonal triads calculated as in point *1 have to be transformed (see Figure 9). This is done according to the following rule:

ITRTRI(IFIG,1/2/3) =
- 1 $v_{1/2/n(new)}$ for conf. IFIG = $v_{1(old)}$ for conf. 1
- 2 $v_{1/2/n(new)}$ for conf. IFIG= $v_{2(old)}$ for conf. 1
- 3 $v_{1/2/n(new)}$ for conf. IFIG= $v_{n(old)}$ for conf. 1
- -1 $v_{1/2/n(new)}$ for conf. IFIG= $-v_{1(old)}$ for conf. 1
- -2 $v_{1/2/n(new)}$ for conf. IFIG= $-v_{2(old)}$ for conf. 1
- -3 $v_{1/2/n(new)}$ for conf. IFIG= $-v_{n(old)}$ for conf. 1.



original triad modified triad

triad modification data for an element of configuration 2

ITRTRI(2,1) = 2
ITRTRI(2,2) = -3
ITRTRI(2,3) = -1

Figure 9: Transformation of the orthogonal triads when using different structural components

Attention: Configuration 1 always represents the shell surface (for which the orthogonal triads calculated as described under point 1* are correct).

*3 These vectors define the crack tip coordinate system that is used to calculate the stress intensity factors and the elastic T-term (Figure 10). The strains in these elements are calculated with respect to this crack tip coordinate system before transforming them to the global coordinate system.

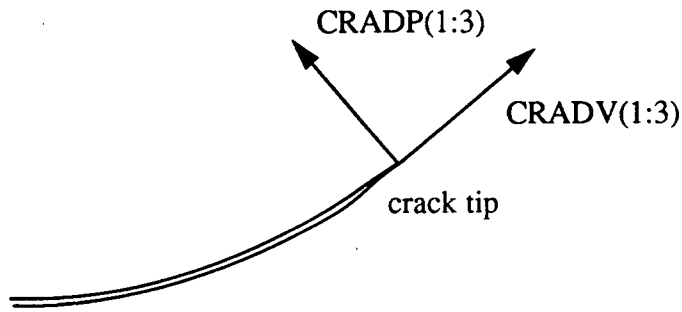
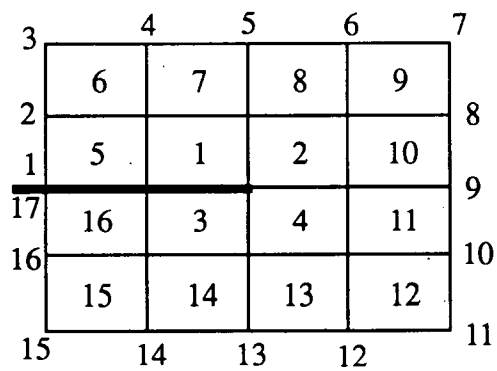


Figure 10: Definition of the crack tip coordinate system

*4 An example for the definition of the enriched elements, the bordering nodes and the integration order is given below (Figure 11).



NCRITM=16 INTM=10

IELENM : 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

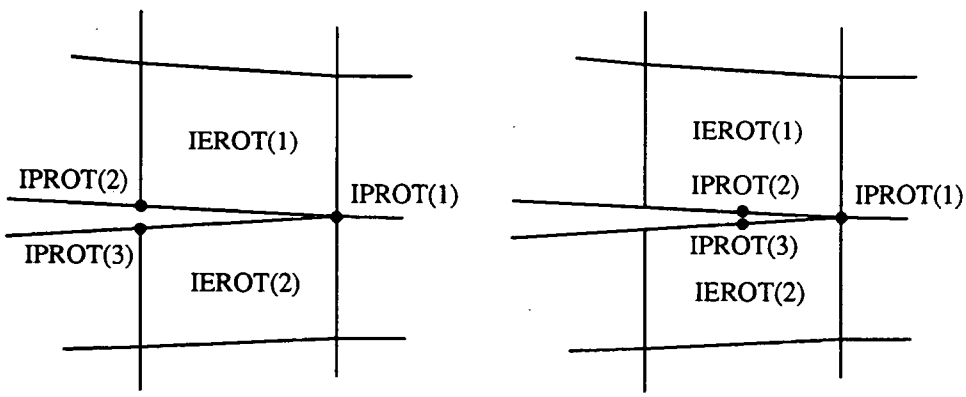
NPBRDM=17

IPBORDM : 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

```
card 13: NCRITM INTM                card 17: NPBRDM
card 14: IELENM(1:16)               card 18: IPBORDM(1:17)
```

Figure 11: Definition of the input data for the enriched element zone used for the calculation of the membrane stress intensity factors (the same type of definition also holds for the bending and tearing stress intensity factors)

*5 These corrections are due to the fact that the in-plane strains according to the analytical crack tip displacement fields will go to infinity when approaching the crack tip. Because of the quadratic terms in the Green-Lagrange strains these large analytical strains close to the crack tip will give a considerable erroneous contribution to the total in-plane strains. This error can be corrected for by subtracting the analytical in-plane strains from the total strains before calculating the stresses (see section 2.1.1.3). However, in order to get hold of these analytical in-plane strains the orientation of the crack tip zone after rotation has to be determined. For this purpose the elements IEROT(1:2) and nodes IPROT(1:3) are used. These elements and nodes have to be defined as pictured in Figure 12.



4-node Bathe element 8- or 9-node Bathe element

Figure 12: Definition of the elements IEROT(1:2) and the nodes IPROT(1:3)

*6 The nodes and elements along the crack faces are used to calculate the elastic T-term via a least square fit technique (see section 2.1.2). The nodes NODEUP(1:NUMNOD) and NODELW(1:NUMNOD) and the elements IELEUP(1:NUMNOD) and IELELW(1:NUMNOD) are defined as pictured in Figure 13.

2.3.2 Input file 'config'

Information concerning the configuration number that the separate elements belong to (see point 1* and 2* in Section 2.3.1) is given in input file 'config'. The 2 input cards that are needed in this input file are as follows:

card	*description*	*format*
1	NFIG : number of element configurations involved in the total mesh NELE : total numer of elements	2*I5
2	IELE : element identification number	NFIG times 2*I5

NUMNOD=7 LSQ1=3

IPOL=2 LSQ2=6

```
card 23: NUMNOD LSQ1 LSQ2 IPOL
card 24: NODEUP(1) NODEUP(2) NODEUP(3) NODEUP(4) NODEUP(5) NODEUP(6) NODEUP(7)
card 25: IELEUP(1) IELEUP(1) IELEUP(2) IELEUP(3) IELEUP(4) IELEUP(5) IELEUP(6)
card 26: NODELW(1) NODELW(2) NODELW(3) NODELW(4) NODELW(5) NODELW(6) NODELW(7)
card 27: IELELW(1) IELELW(1) IELELW(2) IELELW(3) IELELW(4) IELELW(5) IELELW(6)
```

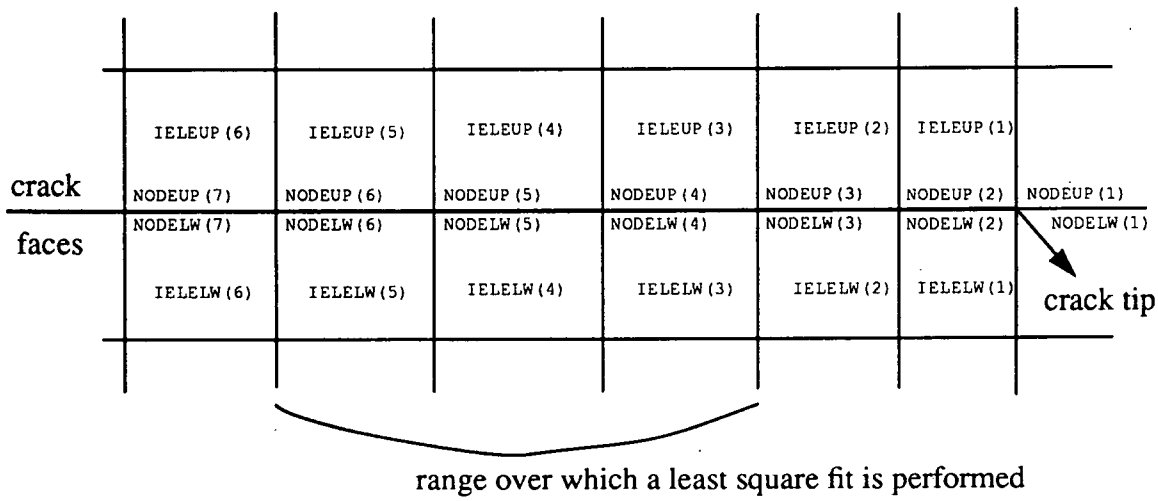


Figure 13: Definition of the input data needed for calculating the elastic T-term

IFIG(IELE) : configuration number that element IELE
IELE belongs to
attention: The element numbering may not cantian
any 'holes' and must be available in ascending order.

3 Error calculation

3.1 Theoretical background

3.1.1 The energy norm

The Z^2 error estimator (or energy norm) is defined as [29]

$$\|e\| = \left(\int_V e_\sigma^T D^{-1} e_\sigma dV\right)^{1/2} \tag{3.1}$$

where

$$e_\sigma = \sigma - \underset{\sim}{\sigma} \tag{3.2}$$

In equation (3.1) D^{-1} is the inverted material stiffness matrix and V denotes the volume of the finite element model. The finite element stresses are represented by $\underset{\sim}{\sigma}$ (equation (3.2)) while σ is the still unknown (close) approximation of the exact continuous stress field.
One way to approximate the exact stress field is by assuming the same interpolation functions for this stress field σ as for the finite element displacement field $\underset{\sim}{u}$. This implies that the interpolation polynomials used for the approximate stress field σ are one order higher than the ones used for the finite element stresses $\underset{\sim}{\sigma}$ (which are described by the derivatives of the interpolation functions for $\underset{\sim}{u}$). The nodal values of the stress field σ are determined via a least square fit with respect to the finite element stress field $\underset{\sim}{\sigma}$. So if

$$\underset{\sim}{u} = N^i u_i \tag{3.3}$$

we now assume that

$$\sigma = N^i \sigma_i \tag{3.4}$$

where σ_i represents the discrete nodal values of the still unknown continuous stress field σ [29]. Next an error E_σ in the stresses is introduced as [3]

$$E_\sigma = \int_V (\sigma - \underset{\sim}{\sigma})^2 dV \tag{3.5}$$

Working out equation (3.5) yields

$$E_\sigma = \frac{1}{2}\int_V \sigma^2 dV - \int_V \sigma \underset{\sim}{\sigma} dV + \frac{1}{2}\int_V \underset{\sim}{\sigma}^2 dV \tag{3.6}$$

Substitution of equation (3.4) into equation (3.6) yields for each element

$$E_{\sigma_e} = \frac{1}{2}\int_V \sigma_i^T N^{i^T} N^i \sigma_i dV - \int_V \sigma_i^T N^{i^T} \underset{\sim}{\sigma} dV + \frac{1}{2}\int_V \underset{\sim}{\sigma}^2 dV \tag{3.7}$$

or

$$E_{\sigma_e} = \frac{1}{2}\sigma_i^T B_e \sigma_i - \sigma_i^T f_e + A_e \tag{3.8}$$

where

$$B_e = \int_V N^{i^T} N^i dV \tag{3.9a}$$

$$f_e = \int_V N^{i^T} \underset{\sim}{\sigma} dV \tag{3.9b}$$

$$A_e = \frac{1}{2}\int_V \underset{\sim}{\sigma}^2 dV \tag{3.9c}$$

Summation of the contribution E_σ over all the elements yields for the total error

$$E_\sigma = \frac{1}{2}\sigma^T B \sigma - \sigma^T f + A \tag{3.10}$$

This error E_σ has to be minimized with respect to σ, thus

$$\frac{\partial E_\sigma}{\partial \sigma} = 0 \tag{3.11}$$

or

$$B\sigma = f \tag{3.12}$$

where B and f were already defined in equations (3.9a-b). The continuous stress field σ is now obtained by solving equation (3.12). This way of working can be applied for all stress components.
Next the energy norm per element can be calculated as

$$\|e_e\| = \left(\int_V e_{\sigma_e}^T D^{-1} e_{\sigma_e} dV\right)^{1/2} \tag{3.13}$$

For the total energy norm this yields

$$\|e\|^2 = \sum_e \|e_e\|^2 \tag{3.14}$$

Although the absolute value of the energy norm has little physical meaning the relative percentage error is more easily interpreted.

$$\eta = \frac{\|e\|}{\|U\|} x\, 100\% \tag{3.15}$$

where

$$\|U\| = \left(\int_V \sigma^T D^{-1} \sigma dV\right)^{1/2} \tag{3.16}$$

and represents the square root of two times the total strain energy. Equation (3.15) is the so-called weighted RMS (root mean square) percentage error in the stresses. The percentage error η is calculated for the whole domain and for all elements separately. The energy norm for the separate elements gives the contribution of each element to the total error and local mesh refinement is performed on the basis of these element errors.
A very common requirement is to specify a maximum percentage error for the total energy norm. One thus requires that after the final analysis is completed the condition

$$\eta \leq \bar{\eta} \tag{3.17}$$

is satisfied for the whole domain, where $\bar{\eta}$ is the maximum permissible percentage error. Equation (3.17) can be translated into placing a limit on the error in each element. One can require, for instance, that the total error is equally distributed over all the elements. This yields for the permissible error for each element

$$\|e_e\| \leq \frac{\bar{\eta}\|U\|}{100\sqrt{m}} = e_m \tag{3.18}$$

where m denotes the total number of elements. According to equation (3.18) the ratio

$$\xi_i = \frac{\|e_e\|}{e_m} > 1 \tag{3.19}$$

defines the elements to be refined and its value can decide the degree of subdivision or the element size needed. Assuming, for instance, the rate of convergence to be $O(h^p)$ the required element size h_r should be

$$h_r = \frac{h_i}{\xi_i^{1/p}} \tag{3.20}$$

where h_i is the current element size. Generally p is assumed to be equal to the polynomial order of the shape functions used [29]. Equation (3.19) also implies that elements with an error $\|e_e\|$ much smaller than the required error e_m can be enlarged.
In contrast to equation (3.18) one can also require the error to be equally distributed over the total volume of the model. This will yield for the maximum permissible error per element

$$\|e_e\| \leq \frac{\bar{\eta}\|U\|\sqrt{V_e}}{100\sqrt{V_t}} = e_m \tag{3.21}$$

where

V_e denotes the volume of element e and
V_t represents the total volume of all the elements.

Compared to equation (3.18) the maximum permissible error is proportional now to the volume of the elements [6].
Both these error distribution options are implemented in the B2000 processor 'b2error' and can be selected interactively by the user. After the calculation of the global percentage error the user can enter a required percentage error that is applied to calculate the reduction factors for all the elements separately. These reduction factors are used by the automatic mesh generator to build a new updated finite element model.

3.1.2 Modified version of the energy norm

A potential disadvantage of the error estimator described in the previous section is that it is difficult to apply to complex structures involving discontinuities. These discontinuities can exist of shell intersections, discrete stiffeners, material- or thickness jumps and/or concentrated forces. For such problems the model has to be partitioned at such discontinuity boundaries because of the stress and strain discontinuities that exist at these boundaries.
Another way to deal with problems involving physical discontinuities is to use an error estimator based on the strain energy (rather than stress or strain). The strain energy namely has to be continuous over these discontinuous boundaries. This alternative also has the advantage of being relatively inexpensive because only one scalar quantity (the strain energy density) has to be dealt with instead of a tensor quantity (stress or strain).
The continuous stress field σ and strain field ε are obtained from the nodal stresses and strains via the shape functions N^i (equation (3.4)). Because the strain energy density U_s is proportional to the product of σ and ε it is obviously inconsistent to use the same shape functions for U_s as for σ and ε. To correct this Stanley et. al. [24] suggested to interpolate $\sqrt{U_s}$ instead of U_s thus

$$\sqrt{\underset{\sim}{U}_{s_e}} = N^i \sqrt{U_{s_i}} \tag{3.22}$$

and this approach seems reasonable.
The energy norm defined in the previous section is

$$\|e\| = \left(\int_V (\sigma - \underset{\sim}{\sigma})^T D^{-1} (\sigma - \underset{\sim}{\sigma})\, dV\right)^{1/2} \tag{3.23}$$

Continuing as above one can substitute $\sqrt{U_s}$ for σ yielding

$$\|e_s\| = \left(\int_V (\sqrt{U_s} - \sqrt{\underset{\sim}{U}_s})\,(\sqrt{U_s} - \sqrt{\underset{\sim}{U}_s})\,dV\right)^{1/2}$$

$$= \left(\int_V (\sqrt{U_s} - \sqrt{\underset{\sim}{U}_s})^2\,dV\right)^{1/2} \qquad (3.24)$$

It is obvious that in equation (3.24) the stiffness matrix D is incorporated in the strain energy densities.

Stanley et. al. [24] proved this error estimator (eq. 3.24) to converge to the exact error for a decreasing element size

$$\lim_{h \to 0} \frac{\|e_s\|}{\|e_{ex}\|} = 1 \qquad (3.25)$$

Further the error estimator of equation (3.24) is a lower bound for the Z^2 error estimator (eq. 3.23).

It is this modified version of the energy norm that we implemented in the B2000 processor 'b2error'. It determines the energy norm after making a least square fit over the strain energy densities in the integration points. These strain energy densities in the integration points are calculated in 'b2cont(crack)' and are written to database file ESED.ibr.0.ityp.0.

3.2 Interactive input commands

The processor 'b2error' calculates the energy norm for a finite element model consisting of Bathe elements only. A further limitation is that the elements are of one type only, 4- , 8- or 9-node. The user has to enter some computational databases and some information concerning the wished type or error calculation. This information has to be entered interactively by answering some questions.

Question 1 : `Enter the archival database used in the nonlinear calculation ?`

Answer 1 : The name of the archival database that was used in the nonlinear calculation with 'b2cont' or 'b2contcrack' has to be entered.

Question 2 : `Enter the archival database with the updated degree of freedom numbers ?`

Answer 2 : In order to perform a least square fit over the complete model one has to solve a linear system of equations where each node has 1 degree of freedom only (namely the strain energy density). This database has to be created by using the B2000 input file where only the first degree of freedom for each node is free.

Question 3 : `Enter the computational database used in the nonlinear calculation ?`

Answer 3 : The name of the computational database that was used in the nonlinear calculation with 'b2cont' or 'b2contcrack' has to be entered.

Question 4 : `Enter the computational database used for the error calculation ?`

Answer 4 : The name of the computational database that is used to save the intermediate information during the error analysis has to be entered. This database is created automatically by the program and thus may not yet exist.

Question 5 :
```
Equal distribution of the error over:
- all the elements  ENTER "1"
- the total volume  ENTER "2"
```

Answer 5 : The type of error distribution has to be selected by just typing 1 or 2 (see also section 3.2.1, equation 3.18 and 3.21).

After entering the above information the program will calculate the local and global errors and the total percentage error. This information will be displayed on the screen followed by the question to enter the wished percentage error, thus:

```
*** T O T A L   E R R O R = ?????
*** T O T A L   E N E R G Y = ?????
*** P E R C E N T A G E   E R R O R = ?????
```

Question 6 : `Enter the required percentage error ?`

Answer 6 : The required global percentage error has to be entered.

This information is used to calculate the element reduction factors that are needed by the mesh generator to refine (or coarsen) a finite element mesh.

3.3 Additional in- and output files

3.3.1 Output file 'ERRORS'

The processor 'b2error' creates an output file 'ERRORS' in which the results of the error calculation are printed. This file contains for instance the strain energy per element, the element errors, the total energy, error and percentage error and the element reduction factors. These reduction factors are needed by the QUADMESH finite element mesh generator [16] for updating a finite element mesh. For this purpose output file 'ERRORS' of routine 'b2error' has to be copied to input file 'reduction' for QUADMESH.
This format of output file 'ERRORS' is as follows.

```
*** E R R O R   C A L C U L A T I O N ***

ELEMENT        STRAIN ENERGY          ERROR
  1                 ?????              ?????
  2                 ?????              ?????
  3                 ?????              ?????
  ..                .......            .......


*** T O T A L   E R R O R = ?????
*** T O T A L   E N E R G Y = ?????
*** P E R C E N T A G E   E R R O R = ?????

REQUIRED ELEMENT REDUCTION FACTORS TO FULFILL A PERC. ERROR OF
????

ELEMENT      RED. FACTOR
  1                ?????
  2                ?????
  3                ?????
  ..               .......
```

4 Solution data mapping

4.1 Theoretical background

4.1.1 Node projection

The solution data mapping routine starts by locating the position of the new nodes with respect to the old element mesh. This position is determined via a projection procedure and the solution data are mapped from the old element mesh onto the new one via the obtained finite element shape functions.

In the projection process the nodes of the new mesh are projected by dealing with the new elements in a sequential manner. The advantage of this approach is that one can always check whether the projection points of the nodes of a new element are situated in and around one and the same element of the old mesh. This way of working will prove to be very helpful in the case of a structure with two or more boundaries that are almost coincident, as for example in the case of a cracked plate.

First a so-called internal node, i.e a node that is not coincident with the boundary of the model, is located for each new element. This procedure is followed to ensure that the projection of the

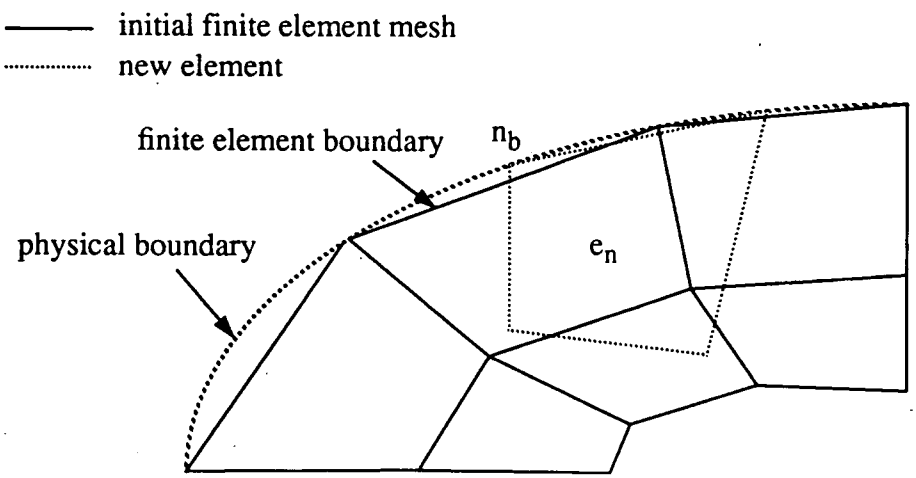


Figure 14: The projection of node n_b of new element e_n is situated outside the old element mesh

first node of each new element does not fall outside the old element mesh (Figure 14). Subsequently, one has to localize the node n_o of the old mesh which is closest to this internal node n_n and also to identify an element e of the old mesh that node n_o belongs to (Figure 15). Node n_n is projected onto the surface described by element e and the ξ_1- and ξ_2-coordinate of this projection point P are checked to see whether this point is situated inside element e. If, for instance, it turns out that $\xi_1 < -1.0$ the projection process must be repeated with the element in the old mesh that has side 1-4 in common with the previous element e (see Figure 16). This technique of projecting node n_n and checking the size of the parameters ξ_1 and ξ_2 must be continued until ξ_1 and ξ_2 lie between -1.0 and 1.0 .

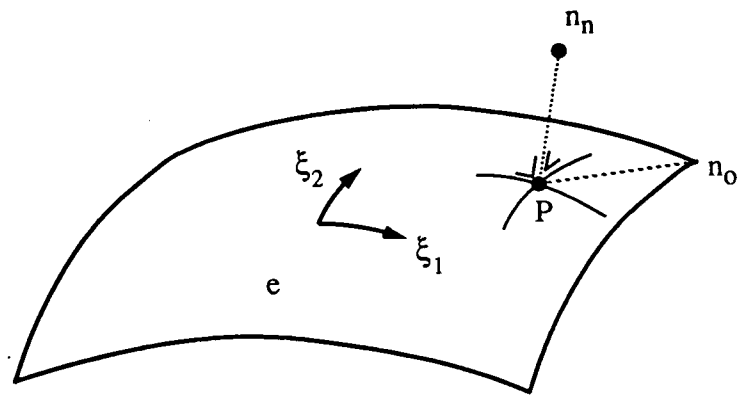


Figure 15: Projection of node n_n onto element e of the initial mesh

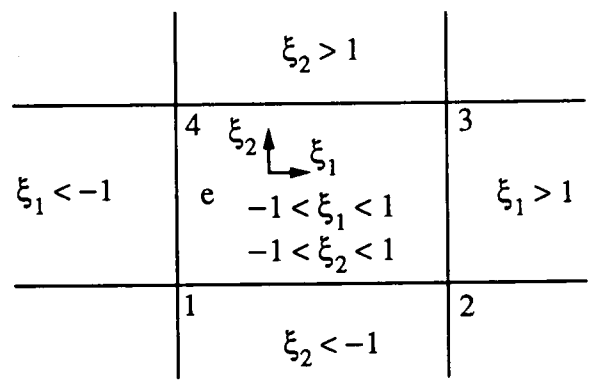
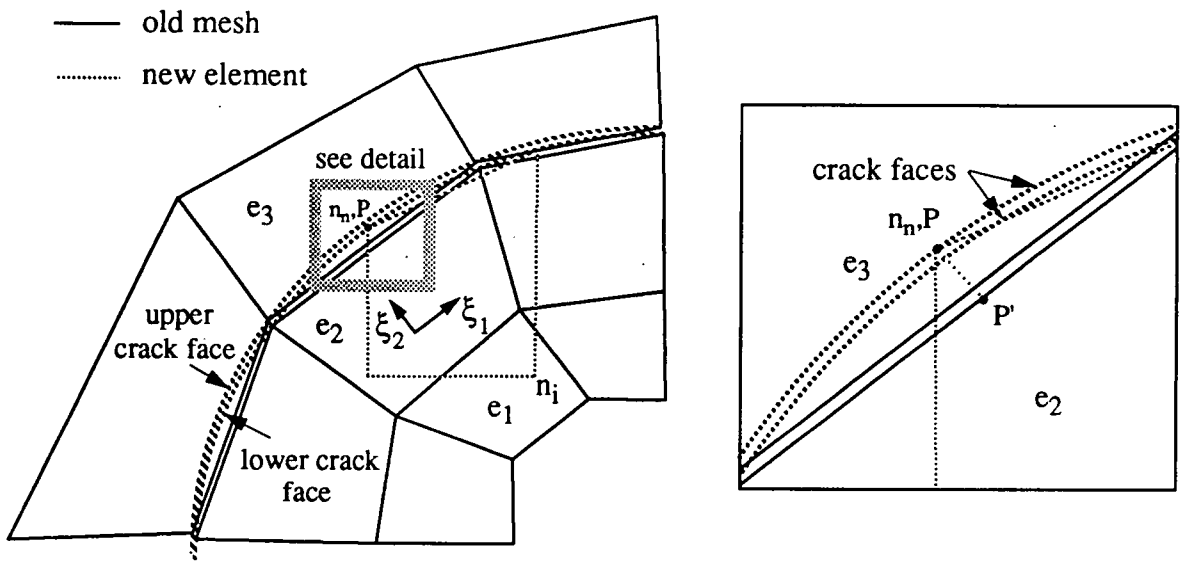


Figure 16: Values of the parameters ξ_1 and ξ_2 in and around element e

The positioning of the remaining nodes of the new element is carried out with the understanding that the projection process is always started from the old element e that contains the projection of the first new (internal) node. The reason for this can best be explained by means of an example.

Suppose that one has two finite element meshes for a cracked plate consisting of 4-node elements (Figure 17). P, the projection of the new node n_n which is part of the lower crack



a. Finite element mesh along a curved crack b. Detail

Figure 17: Node projection for a model with a curved crack

face, is geometrically situated in element e_3 at the upper crack face of the old mesh (Figure 17a). As a consequence the mapped displacements and orthogonal triads for point P on the lower crack face, will be dictated by the corresponding solution data of some old nodes at the upper crack face and it is clear that this will yield incorrect results.
Starting the projection process for node n_n from element e_1, the element that contains the projection of the first (internal) node n_i, one will end up with element e_2 to be the best choice for projecting node n_n on. Due to the curvature of the crack faces however the projection point P is not situated inside element e_2. Therefore P is shifted to P' by setting the value of ξ_1 and/or ξ_2 that is/are outside the interval [-1.0,1.0] equal to the closest boundary of this interval. If, for instance, the final values of ξ_1 and ξ_2 are 0.5 and 1.05 respectively than ξ_2 is set equal to 1.0 (Figure 17b).

4.1.2 The mapping process

The mapping process for the deflections and the base vectors of the orthogonal triads (describing the rotations in the element nodes) from the old finite element mesh onto the new one is straightforward once the position of the new nodes is determined with respect to the old mesh. The coordinates of a projection point P were obtained as

$$(x, y, z)_p = N^i(\xi_{1_p}, \xi_{2_p})\ (x, y, z)_i \qquad (4.1)$$

where

$(x, y, z)_p$ are the coordinates of the projection point P,
ξ_{1_p}, ξ_{2_p} are the parametric coordinates of P,
N^i denotes the corresponding shape functions and
$(x, y, z)_i$ are the nodal coordinates of the element of the old mesh in which P is situated.

The deflections and orthogonal triads in this projection point P (and thus in node n_n of the new mesh) can be determined now in the same way yielding

$$(\mathbf{u}, \mathbf{n})_p = N^i(\xi_{1_p}, \xi_{2_p})\ (\mathbf{u}, \mathbf{n})_i \tag{4.2}$$

where

$(\mathbf{u}, \mathbf{n})_p$ are the deflections and triads corresponding to the projection point P and
$(\mathbf{u}, \mathbf{n})_i$ are the deflections and triads of the nodes of the old element that contains P.

As already mentioned in the introduction, the solution data that are mapped do not correspond to an equilibrium state for the new finite element mesh. However, when the difference between the new and the old equilibrium state is small enough, this new equilibrium state can be obtained within a few iterations. If this difference is 'too large' a load relaxation procedure has to be used whereby the residual force vector resulting from the mapped solution is used as an external load [21].

4.2 Interactive input commands

The processor 'b2intp' maps the solution data from an 'old' finite element mesh onto a 'new' one. Till now the solution data mapping technique is only implemented for the 4-, 8- and 9-node Bathe elements. The user has to enter some information concerning the names of the databases involved, the last load cycle for the nonlinear calculation and the number of structural configurations. This information will be asked for automatically by the program.

Question 1 : `Enter the archival database for the 'new' nonlinear calculation ?`
Answer 1 : The name of the archival database corresponding to the 'new' finite element mesh has to be entered.

Question 2 : `Enter the computational database from which to read the solution data ?`

Answer 2 : The name of the computational database that was used in the nonlinear calculation with 'b2cont' or 'b2contcrack' has to be entered.

Question 3 : `Enter the last load cycle number ?`

Answer 3 : The solution data can only be mapped from the 'old' finite element mesh onto the 'new' one for the last load cycle because the updated triads in the element nodes are available only for this last load cycle. Note that the displacements have to written in branch local coordinates (i.e. DISP.1.icycle). In version 1.77 of B2000 this branch displacement vector has to be generated by running the 'b2ebv' processor.

Question 4 : `Is the number of configuration larger than 1 ? (enter "0" for "no" or "1" for "yes")`

Answer 4 : If one is dealing with a model consisting of different structural configurations 'b2intp' has to take care of the fact that the mapped solution data for each 'new' element are obtained from 'old' elements of the same structural configuration. This is especially important for the orthogonal triads in the element nodes, because the definition of these triads depends on the shape of the separate elements that the same node is part of.

This information is used to map the solution data from the 'old' finite element mesh onto the 'new' one. The mapped displacements are written automatically to the new branch displacement vector DISP.1.0 and to the global displacement vector DISP.GLOB.0. This global displacement vector is used as an initial estimate to restart the nonlinear calculation. Further the nodal triads in the original and the deformed state are written to TRI0.iel and TRIA.iel respectively. These initial displacements and orthogonal triads are entered in the nonlinear calculation by using the option NMESH (see section 2.2).

4.3 Additional in- and output files

4.3.1 Input file 'mode_enrich'

This input file, which is described already in section 2.3.1, is used only to check whether the 'old' solution data are expressed in a cartesian or a cylindrical coordinate system (i.e. card 1, 2

and 3). The mapped displacements will be defined with respect to this same coordinate system.

4.3.2 Input file 'config'

This file contains the configuration data for the '**new**' finite element mesh as explained in section 2.3.2.

4.3.3 Input file 'configo'

This file contains the configuration data for the '**old**' finite element mesh as explained in section 2.3.2.

5 Modifications for the load processor

5.1 Theoretical background

The load processor is modified in such a way that it is possible to express the calculated nodal forces with respect to a cylindrical coordinate system. This extension has to be available in order to perform a nonlinear calculation with 'b2cont' or 'b2contcrack' where the displacements and forces are expressed in a cylindrical coordinate system.
Further one has to be able to apply a pressure load to a selected number of elements. For a cylindrical shell, for instance, the external load has to be applied to the shell elements only and not to the stringer and frame elements. This is accomplished by making use of the configuration data in that the external load defined via the load processor is always applied to elements of configuration 1 only.

5.2 Interactive input commands

No additional interactive input commands have to entered.

5.3 Additional in- and output files

5.3.1 Input file 'mode_enrich'

This input file, which is described already in section 2.3.1, is used only to check whether the calculated nodal forces have to be expressed with respect to a cartesian or a cylindrical coordinate system (i.e. card 1, 2 and 3).

5.3.2 Input file 'config'

This file contains the configuration data for the finite element mesh as explained in section 2.3.2. The external load that is applied via the load processor is restricted to elements of configuration 1 only.

6 Examples

Examples concerning the routines discussed in chapter 2 to 5 can be found in [15].
In chapter 2 of this reference some examples concerning the mode enrichment process are dealt with. These problems only make use of the processor 'b2contcrack' described in chapter 2 of this manual. Next the calculation of the elastic T-term is discussed and applied to several classical and geometrically nonlinear crack problems in chapter 3 of [15].
In chapter 4 adaptive mesh refinement is dealt with. For the sample calculations in this chapter the B2000 processor 'b2contcrack' is applied together with the error calculator 'b2error' and, for geometrically nonlinear problems, the displacement mapping procedure 'b2intp'.
Chapter 5 finally discusses the simulation of crack growth in thin-walled structures and the numerical simulations described in this chapter are performed with all the processors discussed in this user manual together with the automatic mesh generator [16] which is able not only to refine or to coarsen an existing finite element mesh but also to update the mesh after crack extension.

Literature

[1] K.-J. BATHE, "*Finite element procedures in engineering analysis*", Prentice Hall Inc., Englewood Cliffs, New Jersey 07632, ISBN 0-13-317305-4, 1982

[2] S. E. BENZLEY, "*Representation of singularities with isoparametric finite elements*", Int. J. for Numer. Meth. in Engrng., Vol. 8 (1974), pp. 537-545

[3] J. BLAAUWENDRAAD and A. W. M. KOK "*Elementenmethode voor constructeurs: deel 1*", Agon Elsevier, Amsterdam/Brussel, ISBN 90-10-10439-7, 1973

[4] A. de BOER, "*B2000: User manual of the nonlinear analysis processor "b2cont"*", NLR CR 91470 L, National Aerospace Laboratory NLR, The Netherlands, December 1991

[5] D. BROEK, "*Elementary engineering fracture mechanics*", Sijthof & Noordhoff, Alphen a/d Rijn, The Netherlands, ISBN 90-247-2580-1, 1986

[6] C. BUGEDA and E. OÑATE, "*New adaptive techniques for structural problems*", in: "Numerical Methods in Engineering", by Ch. Hirsch, Elsevier Science Publishers B.V., ISBN 0-444-89794-1, pp. 13-24

[7] G. E. CARDEW, M. R. GOLDTHORPE, I. C. HOWARD and A. P. KFOURI, "*On the elastic T-term*", in "Fundamentals of deformation and fracture", Proceedings of the Eshelby Memorial Symposium edited by B. A. Bilby, K. J. Miller and J. R. Willis, Cambridge University Press (1985), pp. 465-476

[8] R. O. FOSCHI and J. D. BARRETT, "*Stress intensity factors in anisotropic plates using singular isoparametric elements*", Int. J. for Numer. Meth. in Engrng., Vol. 10 (1976), pp. 1281-1287

[9] R. J. HARTRANFT, "*Improved approximate theories of the bending and extension of flat plates*", in "Mechanics of fracture 3: Plates and shells with cracks", by G. C. Sih, Noordhoff International Publishing, ISBN 90-286-0146-5, 1977, pp. 45-83

[10] R. J. HARTRANFT and G. C. SIH, "*Effect of plate thickness on the bending stress distribution around through cracks*", J. of Math. and Phys., Vol. 47 (1968), pp. 276-291

[11] G. HEPLER and J. S. HANSEN, "*Mixed mode fracture analysis of rectilinear anisotropic plates by high order finite elements*", Int. J. for Numer. Meth. in Engrng., Vol. 7 (1981), pp. 445-464

[12] P. R. HEYLIGER and R. D. KRIZ, "*Stress intensity factors by enriched mixed finite elements*", Int. J. for Numer. Meth. in Engrng., Vol. 28 (1989), pp. 1461-1473

[13] A. P. KFOURI, "*Some evaluations of the elastic T-term using Eshelby's method*", Int. J. of Fract. (1986), pp. 301-315

[14] H. A. J. KNOPS, "*Computation of stress intensity factors by the mode enrichment technique with applications to geometrically nonlinear problems*", Report LR-736, TU Delft, Faculty of Aerospace Engineering, Delft, The Netherlands, 1993

[15] H. A. J. KNOPS, "*Numerical simulation of crack growth in pressurized fuselages*", Ph. D. Thesis, TU Delft, Faculty of Aerospace Engineering, Delft, The Netherlands, 1994

[16] H. A. J. KNOPS, "*Input description for the QUADMESH mesh generator*", Report LR-759, TU Delft, Faculty of Aerospace Engineering, Delft, The Netherlands, 1994

[17] S. MERAZZI, "*B2000: a modular finite element analysis system - A summary*", NLR MEMO SC 85-041 U, National Aerospace Laboratory NLR, The Netherlands, November 1985

[18] S. MERAZZI, "*B2000: Data access methods and database description handbook*", version 1.2, SMR Corporation, P. O. Box 41, CH-2500 Bienne 4, Switzerland, September 1985

[19] C. D. MOTE Jr., "*Global-local finite elements*", Int. J. for Numer. Meth. in Engrng., Vol. 3 (1971), pp. 565-574

[20] D. R. J. OWEN and A. J. FAWKES, "*Engineering fracture mechanics: Numerical methods and applications*", Pineridge Press Ltd., Swansea, United Kingdom, ISBN 0-906674-26-3, 1983

[21] E. RIKS, "*Some computational aspects of the stability analysis of nonlinear structures*", Comp. Meth. in Appl. Mech. and Engrng., Vol. 47 (1984), pp. 219-259

[22] E. RIKS and H. A. J. KNOPS, "*Mode enrichment and the analysis of through cracks in thin walled shells under internal pressure*", Report LR-736, TU Delft, Faculty of Aerospace Engineering, Delft, The Netherlands, 1993

[23] T.-L. SHAM, "*The determination of the elastic T-term using higher order weight functions*", Int. J. of Fract., Vol. 48 (1991), pp. 81-102

[24] G. STANLEY, I. LEVIT, B. HURLBUT and B. STEHLIN, "*Adaptive refinement (AR)strategies for shell structures; Part 1: Preliminary research*", Lockheed Contract Report LMSC #F318483, May 1991

[25] G. STRANG and G. J. FIX, "*An analysis of the finite element method*", Prentice Hall Inc., Englewood Cliffs, New Jersey 07632, 1973

[26] D. M. TRACEY, "*Finite elements for the determination of crack tip elastic stress intensity factors*", Engrng. Fract. Mech., Vol. 3 (1971), pp. 255-265

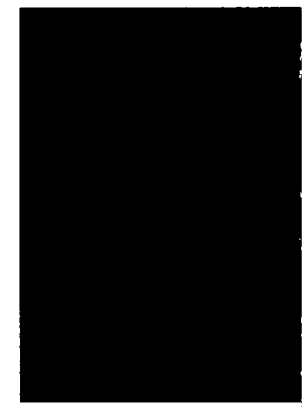
[27] M. L. WILLIAMS, "*The bending stress distribution at the base of a stationary crack*", J. of Appl. Mech., Vol. 28 (1961), pp. 78-82

[28] W. K. WILSON, "*Combined mode fracture mechanics*", Ph. D. Thesis, University of Pittsburgh, 1969

[29] O. C. ZIENKIEWICZ and J. Z. ZHU, "*Mesh regeneration and automatic adaptive analysis*", in: "Computational Methods in Applied Sciences", by Ch. Hirsch, Elsevier Science Publishers B.V., ISBN 0-444-89795 X, pp. 13-24

```
      subroutine AREA(x,y,z,intg,aelem)
      implicit real(a-h,o-z)
      integer intc1,intc2,intc3,intc4,intg(4)
      real x(3000),y(3000),z(3000),l1,l2,l3,l4,l5,
     &     x1,x2,x3,x4,y1,y2,y3,y4,z1,z2,z3,z4,inp1_23,
     &     inp1_34,inp3_12,inp3_14,phi1,the1,phi2,the2,
     &     aelem
      intc1=intg(1)
      intc2=intg(2)
      intc3=intg(3)
      intc4=intg(4)
c     renaming coordinates
      x1=x(intc1)
      y1=y(intc1)
      z1=z(intc1)
      x2=x(intc2)
      y2=y(intc2)
      z2=z(intc2)
      x3=x(intc3)
      y3=y(intc3)
      z3=z(intc3)
      x4=x(intc4)
      y4=y(intc4)
      z4=z(intc4)
c     calculation of lengths of sides and diagonal of quadelement
      l1=((x2-x1)**2+(y2-y1)**2+(z2-z1)**2)**.5
      l2=((x3-x2)**2+(y3-y2)**2+(z3-z2)**2)**.5
      l3=((x3-x1)**2+(y3-y1)**2+(z3-z1)**2)**.5
      l4=((x4-x1)**2+(y4-y1)**2+(z4-z1)**2)**.5
      l5=((x4-x3)**2+(y4-y3)**2+(z4-z3)**2)**.5
c     calculation of enclosed angles
      inp1_23=(x2-x1)*(x3-x1)
     &       +(y2-y1)*(y3-y1)
     &       +(z2-z1)*(z3-z1)
      inp1_34=(x3-x1)*(x4-x1)
     &       +(y3-y1)*(y4-y1)
     &       +(z3-z1)*(z4-z1)
      inp3_12=(x1-x3)*(x2-x3)
     &       +(y1-y3)*(y2-y3)
     &       +(z1-z3)*(z2-z3)
      inp3_14=(x1-x3)*(x4-x3)
     &       +(y1-y3)*(y4-y3)
     &       +(z1-z3)*(z4-z3)
      phi1=acos(inp1_23/(l1*l3))
      the1=acos(inp3_12/(l2*l3))
      if (intg(4).eq.intg(3)) then
c         exception for triangular element
          phi2=0.
          the1=0.
      else
          phi2=acos(inp1_34/(l4*l3))
          the2=acos(inp3_14/(l3*l5))
      endif
c     calculation of area of 3dface as the sum of the area of
c     two triangles
      aelem=.25*(l1**2*sin(2*phi1)+l2**2*sin(2*the1)
     &     +     l4**2*sin(2*phi2)+l5**2*sin(2*the2))
      end

      subroutine WEIGHT(aelem,xmelem)
      implicit real(a-h,o-z)
      implicit integer(i-n)
c     specific weights [kg/mm**3]
      rhocore=0.048E-6
      rholami=1.378E-6
c     number of plies
      n4545=2
      n090=4
c     thicknesses [mm]
      t090= .2114
      t4545=.2696
      tcore=28.6152
c     specific mass of sandwich [kg/mm**2]!
      xm_acore=rhocore*tcore
      xm_alami=n4545*rholami*t4545+n090*rholami*t090
      xm_aelem=xm_acore+xm_alami
c     mass of element [kg]
      xmelem=xm_aelem*aelem
      end
```

```
      program PATCAD
      implicit real (a-h,o-z)
c     ************************************************************
c     PATCAD is an interface program from PATRAN to ACAD12. The
c     input file is a neutral PH-2 file from PATRAN. In this file
c     all cards beginning with other pointers than 1,2 or 99 have
c     to be deleted. The resulting modified neutral file can then
c     be used to obtain a .DXF file which can be used in ACAD12.
c
c                          S. Fransen
c                          29-10-1993
c     ************************************************************
      character in*20,out*20,name1*20
      integer int1,int2,int3,intg(4),count,nrout1
      real x(10000),y(10000),z(10000),atot,aelem,xmass,xmelem
      logical fexis1

      write (*,"('Enter name of inputfile: ',$)")
      read (*,"(a20)") in
      write (*,"('Enter name of outputfile: ',$)")
      read (*,"(a20)") out
      open (unit=1,file=in,status='old')
      open (unit=2,file=out,status='unknown')
c
      nrout1=3
      name1='troep.out'
c
      INQUIRE(FILE=NAME1,EXIST=FEXIS1)
      IF (FEXIS1) THEN
        OPEN(UNIT=NROUT1,FILE=NAME1,STATUS='OLD',
     S       FORM='FORMATTED',ACCESS='SEQUENTIAL')
        REWIND(NROUT1)
      ELSE
        OPEN(UNIT=NROUT1,FILE=NAME1,STATUS='NEW',
     S       FORM='FORMATTED',ACCESS='SEQUENTIAL')
      ENDIF
c
      count=0
      atot=0.
      xmass=0.
      call WRITE0

10    read(1,*,end=100)int1,int2,int3
      if (int1.eq.1) then
          read(1,15)x(int2),y(int2),z(int2)
15        format(3E16.9E3)
          read(1,*)
      else if (int1.eq.2) then
          if (int3.eq.4) then
          count=count+1
          read(1,*)
          read(1,*)intg(1),intg(2),intg(3),intg(4)
          endif
          if (int3.eq.3) then
          count=count+1
          read(1,*)
          read(1,*)intg(1),intg(2),intg(3)
          intg(4)=intg(3)
          endif
      call WRITE1(x,y,z,intg)
      call AREA(x,y,z,intg,aelem)
      atot=atot+aelem
      write(3,*)'atot=',atot
      call WEIGHT(aelem,xmelem)
c     total weight in [kg]
      xmass=xmass+xmelem
      write(3,*)'mass=',xmass
      else if (int1.eq.99) then
      call WRITE2
      write(6,*)count,' QUAD-elements written to ',out
      write(3,*)count,' QUAD-elements written to ',out
      write(6,*)' total area of quad and/or tria elements =',atot
      write(3,*)' total area of quad and/or tria elements =',atot
      write(6,*)'total mass of structure=',xmass
      write(3,*)'total mass of structure=',xmass
      endif
      goto 10


100   continue
      close (1)
      close (2)
      close (3)
      end

      subroutine WRITE0
      write(2,*)'0'
      write(2,*)'SECTION'
      write(2,*)'2'
      write(2,*)'ENTITIES'
      end

      subroutine WRITE1(x,y,z,intg)
      integer intc1,intc2,intc3,intc4,intg(4)
      real x(3000),y(3000),z(3000)
      intc1=intg(1)
      intc2=intg(2)
      intc3=intg(3)
      intc4=intg(4)
      write (2,"('0'/'3DFACE'/'8'/'SHELL'/'62'/i3/
     & '10'/f10.3/
     & '20'/f10.3/'30'/f10.3/'11'/f10.3/'21'/f10.3/
     & '31'/f10.3/'12'/f10.3/'22'/f10.3/'32'/f10.3/
     & '13'/f10.3/'23'/f10.3/'33'/f10.3)")
c     color
      &  150,
c     coordinates corner-nodes of quadelement
      &  x(intc1),y(intc1),z(intc1),
      &  x(intc2),y(intc2),z(intc2),
      &  x(intc3),y(intc3),z(intc3),
      &  x(intc4),y(intc4),z(intc4)
       end

       subroutine WRITE2
       write(2,*)'0'
       write(2,*)'ENDSEC'
       write(2,*)'0'
       write(2,*)'EOF'
       end
```



Rapport 758

60141050442

942570